

Customizable Teaching on Mobile Devices in Higher Education

INAUGURALDISSERTATION
ZUR ERLANGUNG DES AKADEMISCHEN GRADES
EINES DOKTORS DER NATURWISSENSCHAFTEN
DER UNIVERSITÄT MANNHEIM

vorgelegt von

Dipl.-Wirtsch.-Inf. Daniel Schön
aus Simmern/Hunsrück

Mannheim, 2016

Dekan: Professor Dr. Heinz Jürgen Müller, Universität Mannheim
Referent: Professor Dr. Wolfgang Effelsberg, Universität Mannheim
Korreferent: Professor Dr. Ulrike Lucke, Universität Potsdam

Tag der mündlichen Prüfung: 13. Oktober 2016

Abstract

Every teacher struggles with the student's attention when giving a lecture. It is not easy to meet every single student at its knowledge level in a seminar group, and it becomes nearly impossible in huge university lectures with hundreds of students. With the spread of mobile devices among students, Audience Response Systems (ARS) proved as an easy and cheap solution to activate the audience and to compare the students' real knowledge base with the lecturer's estimation. Today, lecturers are able to choose between a huge variety of different ARSs. But as every lecturer has a very individual teaching style, he or she is not yet able to create or customize their individual audience response teaching scenario within a single system. The available systems are quite similar to each other and mostly support only a handful of different scenarios.

Therefore, this work identified the abstract core elements of ARSs and developed a model to create individual and customizable scenarios for the students' mobile devices. Teachers become able to build their individual application, define the appearance on the students' phones in a scenario construction kit and even determine the scenario's behavior logic. Two ARS applications were implemented and used to evaluate the model in real lectures for the last four years. A first ARS was integrated into the university's learning management system *ILIAS* and provided lecturers with basic question functionalities, whereas a second and more advanced stand-alone version enabled lecturers to use personal scenarios in a variety of lecture settings. Hence, scenarios like quizzes, message boards, teacher feedback and live experiments became possible. The approaches were evaluated from a technical, student and lecturer perspective in various courses of different areas and sizes. The new model showed great results and potential in customization, but the implementation reached its limits as it lacked in performance scalability for complex scenarios with a large amount of students.

Zusammenfassung

Viele Dozenten kennen die Situation, dass die Aufmerksamkeit der Studierenden im Laufe einer Vorlesung immer weiter schwindet. Es ist schon nicht einfach die einzelnen Studierenden einer Seminargruppe auf ihren individuellen Wissensständen abzuholen. Mit steigender Größe der Veranstaltung wird diese Aufgabe noch weiter erschwert. Je weiter man sich jedoch von den jeweiligen Wissensständen entfernt, desto schneller schwindet die Aufmerksamkeit der Studierenden. Mit der zunehmenden Verbreitung von mobilen Endgeräten haben sich Audience Response Systems (ARS) als eine einfache und kostengünstige Möglichkeit herausgestellt die Studierende zu reaktivieren und deren Kenntnisse mit der Einschätzung des Dozenten abzugleichen. Heutzutage können Dozenten daher aus einer großen Menge an verschiedenen ARS auswählen. Viele Dozenten haben jedoch einen ganz individuellen Lehrstil welcher trotz dieser Vielzahl oft nicht vollständig mit den vorhandenen Systemen abgebildet werden kann. Sie würden gerne Kleinigkeiten an den vorhandenen Anwendungen ändern oder erweitern. Meist ist das jedoch nicht möglich, bzw. nur mit hohem Programmieraufwand verbunden.

In dieser Arbeit wurden daher die Kernelemente von ARS abstrahiert und zu einem neuen Modell zusammengeführt, so dass individuelle und konfigurierbare Lehrszenarien für die Smartphones der Studierenden möglich wurden. Dadurch wurden Lehrkräfte in die Lage versetzt ihre individuelle Lehranwendung zu definieren und sowohl das Erscheinungsbild für Studierende, als auch die Hintergrundlogik selbst zu gestalten. Im Laufe dieser Arbeit wurden zwei Systeme implementiert, um das Modell in realen Lehrveranstaltungen über vier Jahre hinweg zu evaluieren und neue Erkenntnisse zu gewinnen. Ein erstes, einfaches ARS wurde als Plug-In für die Lernplattform *ILIAS* entwickelt und direkt in die vorhandenen Systeme der Universität eingebunden. Das zweite, darauf aufbauende und höherentwickelte ARS wurde dagegen als alleinstehende Anwendung umgesetzt. Diese ermöglichte den Dozenten persönliche Lehrszenarien in unterschiedlichen Veranstaltungssituationen umzusetzen. So wurden Szenarien wie Quizze, Nachrichtenlisten, Lehrer-Feedback und Echtzeit-Experimente ermöglicht. Die Versuche wurden aus technischer Perspektive und aus der Sicht der Studierenden und Lehrpersonen in Veranstaltungen aus unterschiedlichen Fachgebieten evaluiert und ausgewertet. Dabei zeigte das neue Modell großes Potential im Hinblick auf die individuelle Gestaltungsfreiheit der Lehrszenarien. Jedoch zeigte die Implementierung bei komplexen Szenarien und großen Veranstaltungen mit vielen Studierenden Grenzen in der Geschwindigkeit der Ausführung.

“ [...] it looked like a frontiersman’s shanty thrown together to serve as a mere spring-board for a long flight into the future – a future where so great a field of activity lay waiting that no time could be wasted on the comfort of its start. The place had the brightness, not of a home, but of a fresh wooden scaffolding to shelter the birth of a skyscraper.”

Ayn Rand - Atlas Shrugged

Acknowledgements

I want to thank everybody who was involved in the process of creating this thesis. Be it by directly encouraging my research or by supporting me during the time I worked on it.

First, I want to thank Prof. Dr. Wolfgang Effelsberg for being my doctorate supervisor, for his academic advice and his smart pushes in the right directions,

Prof. Dr. Ulrike Lucke, who agreed to be the second reviewer of this thesis,

Dr. Stephan Kopf and Melanie Klinger for their assistance and collaboration in research and publishing papers,

my colleagues at the department of computer science IV, especially my co-doctoral students Philip Mildner and Philipp Schaber for their steady encouragement and shared suffering,

my superior Ingrid Duda and my colleagues at the university's IT department for covering my back and giving me the time to work on my dissertation,

my parents Gabi and Peter Schön for their support and the starting signals into my academic life,

my brother Florian Schön for keeping my feet on the ground while my head hovered in the clouds,

and last but not least, my beloved Kim Klasen who ensured that the things in my live stayed in the right priority.

Contents

Abstract	iii
Zusammenfassung	v
Acknowledgements	ix
Table of Contents	xi
List of Figures	xv
List of Tables	xix
1 Introduction	1
1.1 Motivation	1
1.1.1 Didactic Challenges	1
1.1.2 Use of Students' Devices	2
1.1.3 Diversity of Application Requirements	3
1.2 Scope of Work	3
1.3 Research Question	4
1.4 The Ideal Scenario	5
1.5 Outline	6
2 Related Work	7
2.1 Educational Background	7
2.1.1 Activation	8
2.1.2 Learning Styles and Flow	9
2.1.3 Teaching Methods	9
2.2 e-Learning	11
2.3 Serious Games	12
2.4 Games with a Purpose	15
2.5 Gamification	15
2.6 Mobile Learning	18
2.7 Audience Response Systems	19
2.8 Generic Approaches for E-learning Systems	24
2.9 Authoring Tools	25
2.10 Learning Analytics	27

3	Model and Design	31
3.1	Concept	31
3.2	Gamification	34
3.3	Basic Model	38
3.3.1	General Conditions	39
3.3.2	Concept	40
3.3.3	Data Model	42
3.3.4	Basic Execution Process	43
3.4	Advanced Model	44
3.4.1	General Conditions	45
3.4.2	Concept	46
3.4.3	Data Model	47
3.4.4	Rules	50
3.4.5	Advanced Execution Process	54
3.5	The Five Phases of a Teaching Scenario	55
3.5.1	First Phase - Blueprint	56
3.5.2	Second Phase - Quiz	57
3.5.3	Third Phase - Interaction	58
3.5.4	Fourth Phase - Result	58
3.5.5	Fifth Phase - Analysis	59
4	Implementation	61
4.1	MobileQuiz	61
4.1.1	Requirements	62
4.1.2	Technology	63
4.1.3	Database Schema	64
4.1.4	Students' View	66
4.1.5	Lecturers' View	67
4.1.6	HomeQuiz Feature	68
4.2	MobileQuiz2	69
4.2.1	Requirements	70
4.2.2	Technology	70
4.2.3	Database	72
4.2.4	Students' View	74
4.2.5	Lecturers' View	76
4.2.6	Scenario Editor	81
4.3	Examples of Use	84
4.3.1	Single and Multiple Choice Questions	84
4.3.2	Likert Scale Evaluation	85
4.3.3	Lecture Feedback	85
4.3.4	Message Board	86
4.3.5	Question Phases	86
4.3.6	Guess Two Thirds of the Average	87
4.3.7	Guess Answer	88
4.3.8	Survey	88
4.3.9	Further Variations	89

5	Evaluation	91
5.1	Technical Evaluation	91
5.1.1	Evaluation of the QR Code	91
5.1.2	Workload on the Network	92
5.1.3	Server Performance in a Classroom Study with the <i>MobileQuiz2</i>	97
5.1.4	Server Performance when Using Various Scenarios	97
5.1.5	Distribution of Students' Smartphones	100
5.2	Evaluation of Acceptance	101
5.2.1	Acceptance by Students	101
5.2.2	Acceptance by Lecturers	118
5.3	Discussion of Results	125
5.3.1	Technical Aspects	125
5.3.2	Students' Point of View	125
5.3.3	Lecturers' Point of View	126
6	Conclusion and Outlook	129
6.1	Conclusion	129
6.1.1	Lightweight ARS	129
6.1.2	Customizable Teaching Scenarios	130
6.2	Outlook	132
	Publications	133
	Supervised Student Theses	135
	References	137
	Abbreviations	160
A	Diagrams	163
B	Survey for the students about the <i>MobileQuiz</i> in 2012	165
C	Survey for the students about the <i>HomeQuiz</i> in 2013	173
D	Survey for the lecturers about the <i>MobileQuiz</i> in 2013	181
E	Survey about the <i>MobileQuiz2</i> in the Fall semester 2015	189

List of Figures

1.1	Picture of a quiz teaching scenario in a seminar of German literature. . . .	6
2.1	Screenshot of an early serious game.	13
2.2	Gamification between game and play.	16
2.3	Distribution of mobile learning devices among students in 2006.	18
2.4	Picture of a visualized question in a famous game show.	20
2.5	Picture of a hardware clicker.	22
3.1	Concept of a quiz application using students' mobile devices.	33
3.2	Low amount of gamification in an ARS.	35
3.3	Screenshot of the game <i>Zork</i>	36
3.4	Screenshot of the game <i>Skyrim</i>	36
3.5	High amount of gamification in the <i>Word Domination</i> Game.	38
3.10	Picture of the board game "The Settlers of Catan".	44
3.11	Elements of the advanced model.	46
3.12	Core data model of the advanced model.	48
3.13	Different contexts of rules in the advanced model.	51
3.14	Processing of rules of different contexts.	52
3.15	Execution process of the advanced model.	55
3.16	Data model of the <i>MobileQuiz</i> in relation to the phases.	56
4.1	Concept of the <i>MobileQuiz</i> implementation.	62
4.2	ER Diagram of the <i>MobileQuiz</i> 's database schema.	65
4.3	Students' view of the <i>MobileQuiz</i>	67
4.4	Lecturers'view of a quiz embedded into ILIAS.	68
4.5	QR code printed on an exercise sheet for self-assessment.	69
4.6	Module structure of the <i>MobileQuiz2</i>	71
4.7	ER diagram of the database schema of the <i>MobileQuiz2</i>	72
4.8	MQ2 AJAX process.	76
4.9	Screenshot of the quiz editor of the <i>MobileQuiz2</i>	78
4.10	Screenshot of the <i>MobileQuiz2</i> 's Game Master view.	79
4.11	Screenshot of the <i>MobileQuiz2</i> 's graphical scenario editor.	83
4.12	Screenshot of a scenario with single and multiple choice questions.	85
4.13	Screenshot of a scenario with Likert scale discussion points.	86
4.14	Listed comments of a Message Board scenario.	87
4.15	Screenshot of the students' view before submitting an answer.	88
4.16	Screenshot of the survey scenario.	89
5.1	Readability of various QR codes.	92

5.2	Network load for one quiz round in bits/s when using the <i>MobileQuiz</i> . . .	93
5.3	Network load during a <i>MobileQuiz</i> usage.	94
5.4	Server connections during a complex <i>MobileQuiz2</i> scenario.	95
5.5	Network traffic on the server during a complex <i>MobileQuiz2</i> scenario. . .	96
5.6	Broken-up time delay of different <i>MobileQuiz2</i> inputs.	96
5.7	CPU usage on the server during a complex <i>MobileQuiz2</i> scenario.	97
5.8	Memory usage on the server during a complex <i>MobileQuiz2</i> scenario. . . .	98
5.9	Load on the server during a complex <i>MobileQuiz2</i> scenario.	98
5.10	Times for object creation in a simple, medium and complex scenario. . . .	99
5.11	Times for rule performance of a medium scenario.	100
5.12	Distribution of operating systems of students in the Fall semester 2015. .	100
5.13	Students' evaluation of the speed of quiz access.	102
5.14	Students' evaluation of the ease of quiz handling.	103
5.15	Evaluation of the <i>MobileQuiz</i> regarding students' interest.	103
5.16	Evaluation of the <i>MobileQuiz</i> regarding students' understanding.	104
5.17	Evaluation of the <i>MobileQuiz</i> regarding students' motivation.	104
5.18	Evaluation of the <i>MobileQuiz</i> regarding regularity of usage.	105
5.19	Students' answers if the course became more interesting.	105
5.20	Students' answers if the <i>MobileQuiz</i> increased the motivation.	106
5.21	Student's answers if the <i>MobileQuiz</i> supported understanding.	106
5.22	Evaluation of the question if the <i>HomeQuiz</i> did increase motivation. . . .	108
5.23	Evaluation of the question if the <i>HomeQuiz</i> increased interest.	108
5.24	Evaluation of the question if the <i>HomeQuiz</i> helped to recognize deficits. .	109
5.25	Evaluation of the question if the <i>HomeQuiz</i> supported participation. . . .	109
5.26	Students' satisfaction when using a quiz with an image.	113
5.27	Students' satisfaction when using a quiz with text only.	113
5.28	Students' satisfaction when all answers were correct or wrong.	114
5.29	Students' satisfaction when they have to enter special characters.	114
5.30	Students' satisfaction in a quiz with an additional progress bar.	114
5.31	Students' satisfaction in a quiz without a progress bar.	114
5.32	Students' answers to the question of increased content understanding. . .	116
5.33	Students' answers to the question if variation increases interest.	117
5.34	Students' statements to the importance of an anonymous quiz participation.	117
5.35	Students' statements to the acceptance of installing a separate software. .	118
5.36	Students' statements to the acceptance of being graded by the system. . .	118
5.37	Number of <i>MobileQuiz</i> -objects in ILIAS.	121
A.1	ER diagram of the full database scheme of the <i>MobileQuiz2</i>	164
B.1	Sheet for the survey about the <i>MobileQuiz</i> in 2012.	166
B.2	Students' answers about the <i>MobileQuiz</i> in economics. (1/2)	167
B.3	Students' answers about the <i>MobileQuiz</i> in economics. (2/2)	168
B.4	Students' comments about the <i>MobileQuiz</i> in economics. (1/2)	169
B.5	Students' comments about the <i>MobileQuiz</i> in economics. (1/2)	170
B.6	Students' answers about the <i>MobileQuiz</i> in German literature. (1/2) . . .	171
B.7	Students' answers about the <i>MobileQuiz</i> in German literature. (1/2) . . .	172
D.1	Survey for the lecturers about the <i>MobileQuiz</i> . (1/3)	182

D.2	Survey for the lecturers about the <i>MobileQuiz</i> . (2/3)	183
D.3	Survey for the lecturers about the <i>MobileQuiz</i> . (3/3)	184
D.4	Results of the survey for the lecturers about the <i>MobileQuiz</i>	185
D.5	Comments of the survey for the lecturers about the <i>MobileQuiz</i> . (1/3) . .	186
D.6	Comments of the survey for the lecturers about the <i>MobileQuiz</i> . (2/3) . .	187
E.1	Lectures' evaluation of the <i>MobileQuiz2</i> customization. (1/2)	190
E.2	Lectures' evaluation of the <i>MobileQuiz2</i> customization. (2/2)	191
E.3	Reluctant lectures' evaluation of the <i>MobileQuiz2</i> customization. (1/2) . .	192
E.4	Reluctant lectures' evaluation of the <i>MobileQuiz2</i> customization. (2/2) . .	193
E.5	Students' evaluation of the <i>MobileQuiz2</i> customization.	194

List of Tables

3.1	Common types of attributes with their final visualization.	49
3.2	Set of conditions used in rules of the advanced model.	53
3.3	Set of actions used in rules of the advanced model.	54
3.4	The five phases with involved roles and locations.	55
5.1	Test devices and their camera resolution.	92
5.2	Test scenarios and their parameters.	98
5.3	Students' reasons for their lack of participation.	102
5.4	Distribution of the students' devices.	102
5.5	Participation in the <i>HomeQuiz</i>	107
5.6	Selected parameters and hypothesis of the students' behavior.	112
5.7	Average answer values of the survey about the scenario editor.	124

Chapter 1

Introduction

1.1 Motivation

Mobile devices are a common sight in university lectures nowadays. The majority of students owns at least one web-enabled smart device like a smartphone, tablet PC or laptop. They are used for communication, organization, information and entertainment. It therefore was a logical step to use them in teaching as well. E-learning is still a growing research field and there are many possible fields of usage for mobile devices.

One of the most efficient opportunities is the usage of smartphones in Audience Response Systems (ARS). this usually requires only a small effort compared to the gained benefit. Beside special hardware clickers, early software implementations of ARSs were written for Personal Digital Assistants (PDAs) and handheld computers to increase the interactivity in the classroom, activate the audience and get a realistic feedback of the students' knowledge. With today's availability of smartphones among students, the usage of ARS in classroom settings gets another stimulus. Phone and web technologies evolved so far that universities were enabled to develop their own systems with reasonable effort.

But with the development of such an application, several aspects have to be taken into account. Didactic goals and methods have to be considered, the platform for the application and devices has to be chosen, the solution should be integrated into the already existing e-learning environment, and it should support the lecturers to perform their individual teaching styles.

1.1.1 Didactic Challenges

Audience Response Systems try to meet several didactic challenges at once. The most quoted ones are the insights about the real knowledge bias of the students and the lack of interaction during lectures. Furthermore, ARSs offer the opportunity to use modern everyday technology to improve established teaching scenarios, especially in large lecture sizes, when an input of many students is desirable.

Many lecturers are faced with the problem of students' insufficient preparation and post-processing of their courses. Students usually give little feedback if asked for questions concerning the current course. They do not dare to ask, afraid to embarrass themselves.

In other cases, they are not able to articulate their gaps in knowledge because they cannot identify them without concrete tasks. As a consequence, the lecturer does not know the real bias between the students' base of knowledge and his own impression of it. One common solution for this is to provide scored homeworks and weekly exercise sheets. This gives students the opportunity to practice and evaluate their knowledge base, and therefore points out their real gaps. The largest drawbacks are the high amount of work to create the weekly sheets and the high efforts to score them all. In large lectures with hundreds of students, this is usually achieved by a decent number of supporting tutors. Audience Response Systems offer the opportunity to get an overview of the students' knowledge with only a fraction of this effort.

As it is quite difficult to stay attentive during a whole ninety minutes lecture, students usually get sleepy and bored after twenty to thirty minutes since the teacher started (see [138]). The motivation to stay attentive further decreases with the progress of the lecture. Performing an audience feedback can reactivate the listeners as they are individually addressed [161]. They have to rethink the content by themselves and may talk about it with their neighbors. They have to submit a personal solution to the current problem and become aware of their real state of knowledge, which otherwise is often overestimated, when they only approve the answers given by others.

In some teaching situations, it is convenient to collect the opinion of students about a given topic. This may be an estimation about progression of a currency in economics, or the attitude towards an ethical question in philosophy. Getting a public picture of the students' opinion can easily be achieved by counting hand signs, but performing an anonymous survey during the lecture can be quite challenging. When discussing a controversial topic, the possibility rises that some students do not dare to stand for their true opinion in public. For this, ARSs can be a solution as anonymous surveys can be performed with a minor effort.

1.1.2 Use of Students' Devices

Lecturers get quick feedback about the state of knowledge, comprehension, and opinions of listeners by using an ARS. Students are able to express their knowledge without stepping forward in front of the entire auditorium. Therefore, *Clicker* systems were invented and merchandised. These clickers had all the necessary technology built into little electronic boxes with a few mechanic answer buttons. But this solution had a set of disadvantages: The voting devices and the corresponding receivers were quite expensive, and the wireless devices had to be distributed to the students before the start of every lecture, and recollected at the end. The acquisition costs, the administrative overhead, and the risk to loose devices thereby increased with larger numbers of students. The content of the quizzes had to be prepared before the lecture started, and the answering options were severely limited by the design of the devices (e.g., multiple-choice questions only). But with the upcoming propagation of smart mobile devices (especially smartphones) among students, the situation provided the possibility to take advantage of these new devices instead of the specialized hardware clickers. Our idea was to exploit the proliferation of smartphones, tablet PCs and laptops among students to design and implement a *MobileQuiz* application. Nowadays, most students own a smartphone or another mobile Web-enabled device like a netbook, laptop, or tablet PC. All these devices have built-in wireless LAN access and thus have the ability to be used as response devices.

1.1.3 Diversity of Application Requirements

Due to the didactic benefit and the propagation of smartphones, the amount of ARS implementations grew significantly in recent years. Many of these applications were quite similar to each other and fitted for the purpose of a few specific classroom learning scenarios, like test questions, self-assessment or simple audience feedback. Within the scope of this thesis an ARS was created in the Fall semester 2011. It was integrated into the already existing learning management system of our university and used the students' smartphones as "clicker" devices. This *MobileQuiz* had then been adapted by many lecturers of our university during the following semesters. But with the increasing popularity, more and more requirements for extensions were brought to our attention. These requirements ranged from simple layout adaptations to enhancements of the number of different question types, up to new learning scenarios with their own, highly specialized logic. Beginning with simple feedback and self-assessment scenarios, the lecturers wanted to have customized teaching scenarios with more complexity, adaptivity and increased student interactivity. They were interested in guessing questions with a range of correct answers, text input for literature discussion, twitter walls for in-class side discussions, game-theory and decision-making experiments for live demonstrations. Every lecturer had a precise picture of his or her requirements that differentiated from the existing scenarios and the requirements of the other lecturers. But meeting the lecturers' needs by adding new question types and other extensions into the existing quiz logic required an additional programming effort for every new scenario. Even small variations in appearance resulted in modifications within the program code. New features like an adaptive and collective quiz behavior would have needed a full reconstruction of the existing application or an entirely new tool for this purpose.

This is probably one of the main reasons why so many different ARS implementations exist right now. A university which is interested in using an ARS has a difficult time to find an application which perfectly fits its requirements. Due to the relatively small effort to create a new ARS, they decide to create their own application. But as before, this new application only fits the requirements of a few other universities. Lecturers soon are seeking for additional functionalities or changes in the already implemented question types.

1.2 Scope of Work

The scope of this thesis focuses on solutions for teaching scenarios in university lectures, exercises and seminars with usual classroom sizes between five and one hundred students. It does not consider environments with hundreds or thousands of people in the audience. The listeners are mainly university students who own and use a modern smartphone or similar mobile device like a tablet PC. The participation in the developed application is anonymous, no authentication or authorization is required. Therefore, grading and other personalized use cases are not supported very well. Furthermore, the general flexibility and the customization of teaching scenarios are focused, rather than the technological and didactic perfection of the individual scenarios.

1.3 Research Question

Regarding the described problems in the defined scope this thesis searches for answers to the following questions:

What are the requirements for a tool that supports the lecturers in understanding and interacting with their students in the classroom?

How can these requirements be met with the current state of technology?

Which bottlenecks and difficulties have to be considered and may not be solved by technology alone?

In an approach to answer these research questions, this work postulates the following hypotheses.

Simple set-up: An application which meets the demands, requires a low access barrier. This does effect lecturers and students equally. Lecturers are not very willing to install new software on their own devices. They also want the used applications to be easy to learn and reliable. Students do not want do install software on their mobile devices either. They prefer applications which can be used without any initial effort.

Integration into existing infrastructure: An application which is fully integrated into the existing infrastructure offers benefits in a short and long time perspective. It lowers the barriers for its first use, as no additional hardware has to be purchased, no new applications have to be learned and no installations have to be done by the lecturers. Lecturers only have to work with extensions of already existing systems. In the long term, the efforts and costs of maintenance are noticeably lower than with proprietary stand-alone solutions. In addition, more lecturers will steadily adapt the new functionalities, because they are frequently confronted with them during their daily work. If a lecturer notices a quiz functionality every time he uploads his weekly script to the e-learning platform, he or she will try the quiz more likely as if a separate system has to be used instead. The disadvantage of dependence on a specific learning management system is therefore accepted in the frame of this work.

Simple access: The quiz application is far more accepted by the students, if the effort to access the individual quiz rounds is fast and simple. An application which supports the students so that they only have to scan a QR code to enter a quiz is thereby preferred over an application at which the access requires the preceding installation of a software, an authentication or the manual input of a URL.

Customization of teaching scenarios: Lecturers are used to predefined teaching scenarios. They usually perform quizzes with common elements like single and multiple choice questions. But when they get used to these standards, they develop new ideas for individual extensions and want to adjust the teaching scenarios more precisely to their current needs. Unfortunately, it can be very expensive to customize already existing functionalities, so lecturers have to stay with the few existing teaching scenarios. Customizable teaching scenarios therefore will enable the lecturers to fully scoop their teaching potential and thereby increase their motivation.

1.4 The Ideal Scenario

When considering an ideal teaching scenario that uses the students' mobile devices, a lecturer would usually start with a general idea. Perhaps, he or she then would discuss this idea with a didactic expert and define the scenario's model and logic. The expert would be able to design the scenario with the help of a graphical authoring tool. The software application would support typical elements like texts, numbers, check-boxes and more specialized ones like progress bars and submit buttons. The scenario's logic would support common logical operators, math operators like arithmetic mean and sum, and visual operators for hiding and displaying elements on the screen. The lecturers would then choose this scenario to fill it with the concrete contents of the lecture. While new scenarios would be available to every lecturer, the concrete quizzes would have restricted rights for the respective lecturer. Lecturers would start a new game round of one of their quiz entities at any time. With the start of a game round, a new QR code would be created and could be used within the classroom or within other media like videos or exercise sheets, as shown in Figure 1.1. The QR code would be easily readable as it is the main connecting point for the audience. Students would then access the application with any Internet-enabled device. The quiz page would only be loaded once, and every change or user action would be sent to the server via an AJAX request. After processing the rules and changes on the server side, the student's mobile browser would get an AJAX response containing all relevant information. The page content and layout would be changed by JavaScript without the need for a separate page refresh. After performing a game round, the results could then be discussed with the audience or used for the preparation of an upcoming lecture. The application would store any meta data so that all kinds of analyses of the students' behavior could be done afterwards. Besides the technical information about the participating devices, the users' activity and the performance of the students would be tracked.

In a more concrete example the scenario could be as follows: A lecturer in game theory (business economics) wants to perform a live experiment. The game is called *Guess $\frac{2}{3}$ of the average*. In this experiment, every participant has to choose a number between 0 and 100. The winner is the person that is closest to $\frac{2}{3}$ of the mean value of all answers. The lecturer first talks to a didactic expert, and they decide to model a scenario with a numeric input field and a submit button. If students click the submit button without any number or with a value higher than 100, a warning text should be displayed, otherwise the button and input field should disappear, and a text message should appear. After a given amount of time, the mean value of all numbers should be calculated together with the corresponding $\frac{2}{3}$ value. The student with the smallest difference between his answer and to this $\frac{2}{3}$ value should see a winning message. The expert now models this scenario within a graphical authoring editor. The lecturer then chooses this scenario



FIGURE 1.1: Picture of a quiz teaching scenario in a seminar of German literature.

within the application to create a quiz entity of it and adds the name of his lecture and some describing text. He or she enters the quiz editor and selects a time span for the students' input. During the lecture, the course page is opened within the university's learning management system, and a new quiz round is started. The students participate in the experiment, and while the winner is happy about his guess, the lecturer discusses an overview of the distribution of all guesses.

1.5 Outline

This thesis is structured as follows: Chapter 2 will capture the current state of research in the field of e-learning and focused on mobile devices. The didactic background will be presented with activation, learning styles and teaching methods. After the didactic theory, gamification, relevant serious games, games with a purpose and audience response systems will be presented. In addition, publications about authoring tools and learning analytics will be introduced. Chapter 3 will present the fundamentals of this thesis. The concept of gamification will be described before the basic model of the structures and mechanics of an audience response system are presented. Section 3.4 then explains the advanced data model of the *MobileQuiz2* with its different phases. Next, Chapter 4 presents the two applications *MobileQuiz* and *MobileQuiz2* which were designed and implemented during the work of this thesis. Furthermore, an implementation of a graphical scenario editor is introduced and various teaching examples on using the *MobileQuiz2* are explained. Chapter 5 finally presents the various evaluations of the tools and their usage. The evaluation of the different technological aspects like the impact on the university's WiFi network and the handling of the QR code are evaluated first. The surveys on students and lecturers are described in Section 5.2.1 and Section 5.2.2. Section 5.3 then discusses the educational and usage results. Finally, Chapter 6 presents a conclusion about the work and depicts possible topics of further research.

Chapter 2

Related Work

The use of technology in education is a popular topic for quite a long time, and increasing engagement and participation of the students with some kind of response system has always been one aspect of educational research. Judson [115] dates back the beginnings of electronically supported learning (e-learning) to the U.S. military, which started to use filmed instruction materials in the 1950s. Seventy years later, we are now living in a fully engineered world, where education without technology would seem archaic.

The next pages give an overview of research and developments of technology in education, which finally led to the situation of using the students' smartphones in university classrooms nowadays. After an introduction to the basic didactic principles in Section 2.1, education and technology come together as e-learning in Section 2.2. Trying to merge fun and learning in the fields of serious games and gamification is explained in Section 2.3 and Section 2.5. With the reduction of the size of the devices, the learners' mobility increased and opened the field of mobile learning described in Section 2.2. The increased mobility of devices then supported the development of Audience Response Systems (ARS) as described in Section 2.7. When trying to expand the functionalities of ARS, generic solutions and lecturer-friendly authoring tools have to be considered, as presented in Section 2.8 and Section 2.9. Finally, learning analytics have to be used to analyze the success of the learning process in Section 2.10.

2.1 Educational Background

“Education plays a crucial role in the overall development of the country and the society” [170]. Teaching and learning should therefore be efficient and effective. Research in educational psychology has shown that content cannot simply be transferred from the lecturer to the learner [33]. Students need to “work” with the given information, comprehend it, make sense of it, relate it to prior knowledge and save it in their memory. Learning can be defined as an active process [108, 208]. Lecturers can support this process by offering their students opportunities to actively deal with the presented learning content. Ramsden [171] has showed that an active involvement of learners in the lecture has a big impact on the success of learning.

2.1.1 Activation

Especially basic introductory courses at universities are large lectures with hundreds of students, which have to be addressed by the lecturer at once [161]. This quantity of students can then have disadvantages with attendance, learning, discipline and general alienation. Lecturers always have to keep the balance between in-class activities and the need to cover the required course content [92]. And although classic face-to-face lectures are very scalable and cost effective [125, 178], the content often is not very self-motivating [168]. Many students get bored very fast by passively attending dry presentations. The usual time frame of the students' attention is around 20 minutes which is quite challenged in a 90 minutes lecture without a pause or diversion [125]. Additionally, many lectures lack real-world references and only focus on facts and memorization, instead of critical analyses and discussion [161]. This is clearly not the best way to teach as motivation has a great impact on the students' performance in learning [125, 168].

Another drawback of the classic teaching paradigm is the point in time when the lecturer gets real feedback about the students' state of knowledge. Except for mid-term assessments, the real feedback usually comes not until the exams at the end of the semester. At this moment, it is too late to correct mistakes and remove ambiguity. Students have no chance to realize their knowledge gaps on time and do not benefit from the awareness of them.

Furthermore, students usually have different learning styles and learning paces, which is not regarded by the classic way of teaching. Therefore, mismatches emerge between the students' learning styles and the lecturers' teaching styles, which further decrease the students' motivation [131]. Lecturers try to motivate and activate students by asking questions about the current content or a content-related topic. They engage students to participate, and they pick-up the knowledge level of the selection of people who actively answer. But this insight might not be representative, especially in large lectures. In many cases, only a few students interact with the lecturer and benefit from the activation [125]. Some students are too shy to participate and are not comfortable with situations where the whole course's focus is pointed on them. These students hold back to not embarrass themselves, for their fear of failure and critics [141]. This effect usually increases with the size of the lecture [144]. The traditional way of teaching does not provide enough opportunities to involve these introverted learners. But if a teacher tried to teach all student well in the classic way, he or she had to fourfold the student contact time [131]. Therefore, educational institutions reviewed their teaching methods in order to come up with new ideas to optimize their lectures in terms of engaging all types of students. Research in the field of educational psychology (e.g., Jonassen [114]) showed that actively involving students can improve this situation considerably. It is not feasible to simply transfer information from the lecturer to the students. Students have to handle the content in order to comprehend and store it in their memory. This process is called "active learning" and shifts the teacher-centered learning towards a student-centered approach [112]. Active learning happens when teachers create learner participation directly into classes with methods that ask students to apply newly acquired knowledge for solving problems [143]. It is a good solution to reduce distraction by keeping students engaged [95], as they became an active part in the learning process. The students learning curve is improved and the creative thinking is inspired, the learners become the center of their own learning environment [141].

One approach to stimulate active learning was to provide the students with possibilities of rehearsing: applying information after hearing it in a lecture enables the connection between concepts, makes the information meaningful, relevant, and contextual to the students [34]. Motivation research showed that diversity can improve the students' motivation to learn. Providing variations in teaching methods can make it easier to concentrate. Offering challenges and opportunities to succeed influences the students' commitment. Possibilities to self-control the learning progress give them a realistic progress notification. External control by the lecturer help students to understand their learning deficits, and informative feedback gives them an additional feeling of being supported and appreciated [69]. Therefore, activation (especially in the sense of rehearsing) and feedback are substantial for the student learning process. Major challenges concerning the use of activating methods and feedback are time constraints and the size of the course. In traditional lecturing, lecturers usually do not have much time to test the students' understanding of a learning unit, especially in courses with hundreds of students.

Critical research noticed that active learning only is another form of classic homework. But Prince states that active learning refers especially to in-class activities [169]. The student activity and engagement during the learning process in presence of the lecturer is the focus of active learning. But as a drawback, researchers like Gehringer and Peddycord [95] argue that active learning takes much more time from lecturing than classic presenting-centered teaching styles.

2.1.2 Learning Styles and Flow

As mentioned in above, different people have different learning styles. A learning style is the way in which information is processed. It describes how a person perceives, conceptualizes, organizes and recalls information [48]. Students have a wide range of different learning styles which are usually not covered by equally different courses with appropriate teaching styles [131]. Lage et al. [131] arrange learning styles into three different systems to classify them: the Grasha-Riechmann Learning Styles Questionnaire [174], the Keirsey-Bates categorization [117], and the categorization according to Kolb [149]. All classifications focus on the individuals' strengths, rather than their weaknesses. Therefore, when the instructor's teaching style does not match the students' learning styles, the students' motivation and learning performance may decrease [131].

When instructors perfectly meet a students learning style, students can reach a status called *flow* [63, 64]. That means that the student is psychologically involved in a certain task, that everything else is blurred out. It is a high state of enjoyment which is also intended to reach in computer and video games [50, 61]. But achieving this state can be quite challenging as every students would have its own "operating point" for the flow.

2.1.3 Teaching Methods

One possible solution to meet the mentioned requirements for active learning in the classroom is the use of quizzes. Quizzes may be a series of questions answered through some kind of hand-held devices. The questions provide additional communication functionalities and thus help to improve the activity and motivation of the students.

Quizzes can be used to ask the students anonymously about their understanding or their opinion. Students can check their understanding of the course content and apply the information in the quiz. Lecturers get an insight in the students' learning progress and possible deficits. This enables lecturers to adapt the explanations and the timing of the lecture according to the students' needs [143]. Besides, students can be asked for their opinion concerning the course content. This method gives more reserved students the opportunity to participate. The results can then be used to start a discussion (as an icebreaker) or to directly ask further questions face-to-face.

Another way to use quizzes is to ask students about their satisfaction with the course. Because motivation increases if students feel involved and appreciated [69], a serious request for their satisfaction can highly influence the atmosphere in the classroom. Furthermore, the results can give important hints on the course concept and teaching activity. The lecturer can use this information to adapt his or her teaching activity, or explicitly state the reasons for the course design (often, the dissatisfaction is caused by a lack of communication and in-transparency of expectations, not by an insufficient course design). The use of activating elements, in means of communication in lectures, can have a positive impact on the learning success of the students, on the classroom atmosphere, and on the satisfaction on both sides.

When activating students with quizzes succeeds, a class discussion can be started. Class discussions are a very efficient technique to share ideas with classmates and create knowledge out of given information. Students can listen to different perspectives on a certain problem, whereas teachers get aware of many aspects on students' communication and problem solving skills. Bruner states that instructors should not produce little walking libraries, but rather get students to think clearly for themselves and take part in the process of knowledge creation [39]. According to him, knowledge is a process rather than a product. Therefore, the exploratory learning process is highly important [183]. But instructors have to be aware of the fact, that the performance of groups is highly dependent on the level on which the individual group members work together, and on how relevant the content is for the audience. The situation has to be suitable to encourage conceptual understanding instead of learning facts only [143].

To face the problem of loosing time during lectures when using active learning methods, so-called *inverted lectures* came into the center of attention. Inverted lectures, inverted classrooms, flipped lectures, or flipped classrooms flip homework and lecture. The main idea is that the content information is provided before the actual "lecture" starts, and it should be processed by the students in preparation of the meeting with the lecturer. During class, students are involved into active learning sessions [92]. The lectures are focused on the interaction among students and the interaction between the students and the lecturer. The content information is usually provided in a technology-supported way in shape of videos, scripts or audio files. In contrast to classic lectures where the information is delivered at a specific point in time, students can access all the available content information at the time and place of their own choice. Furthermore, when providing different kinds of redundant content information, learners can choose the learning method themselves, which optimally fits to their individual learning style. Slower students may be able to repeat selected parts, whereas faster students may skip parts they already mastered. It therefore increases the probability to get them into a flow-like learning state. And despite the high flexibility, the syllabus can still be covered by the (finite) number of in-class meetings. The contact time can be used more efficiently by focusing on actual comprehension difficulties and collaborative learning

activities [95, 133]. Instructors further benefit, as they do not have to teach the same content in a similar, static way in every course again. Every semester brings a new group of students with individual approaches to the material. Therefore, the inverted classroom is also an effective method against teacher boredom [131].

2.2 e-Learning

Being aware of the didactic theories, many approaches were made to support learning and teaching by using modern technology. According to the Encyclopedia of Science Education “E-learning (electronic-learning) refers broadly to the use of information and communication technologies for the electronic delivery of instructional content and the support of educational processes.”[106]. At this, learning can be absolutely self-paced with no human at all, or highly coordinated with a human instructor attending from a distance. Electronic e-learning systems have been developed since the 1960s for many audiences and purposes. Thereby, new applications closely followed new trends in technology. Now, e-learning has manifestations in multimedia classrooms, Internet technologies, videos, mobile devices, MOOCs and interactive whiteboards. E-learning is therefore mostly referred to as the use some kind of electronic devices in combination with technology, such as the Internet or other networks, audio or video, satellite broadcast or interactive TV [104, 184, 204].

Many studies have investigated the effects of different e-learning scenarios compared to traditional learning. But because of the huge field of possibilities, the results are quite heterogeneous. When asked directly, students mostly prefer the human instructor [60, 148] and usually spend less time in web-based learning than in traditional learning environments [30]. When considering the learning outcome, a study from Miley et al. [148] showed no significant differences, whereas Bele and Rugelj [30], Judson [115] and Feng and Heffernan [84] could recognize better learning achievements with e-learning methods. It seems to strongly depend on the specific method and purpose. E.g., Lapuh Bele and Rugelj [132] got better success with audio animations than static picture content whereas Giannakos and Vlamos [96] reached good performing students far less than other students with their Webcast-experiment. Additionally, the success of online instructions depends on student conditions. Courte [60] states that “students must be comfortable with technology and able to easily navigate online materials, or else they will easily feel frustrated and unmotivated”. Also, when unsupervised, the dropout from online courses is much higher than from traditional courses. In conclusion, e-learning can achieve significant better learning results when used correctly.

Although universities and schools have recognized the benefits of integrating e-learning technologies into their local educational settings [81, 164, 204], computerized technology as a tool for interactive learning is not fully used in higher education everywhere [21]. But teaching is no longer restricted to the physical classrooms. The students’ behavior grew with the development of the Internet, and their role shifted from passive content consumers to learning content producers. The Internet helped to make learning more dynamic and user-centered.

Especially popular are so-called *learning management systems* (LMS). They are the key element of most universities’ e-learning systems. According to Edrees [81], almost 75% of the 129 top U.S. universities already used web-based systems in 2007. As universities

have to face an ever increasing number of students, technological support in managing the teaching is a consequent strategy. Learning management systems are complex information systems that manage the content and communication channels between lecturer and students. The students are provided with course content in the form of lecture slides, videos and additional literature. They are provided with communication and collaboration functionalities like wikis, chats and forums. They can test their knowledge with quizzes and track their learning progress on individual project pages [187]. Especially, communication tools offer the kinds of active engagement which is preferred by students of the Web 2.0 generation. They can use these tools according to the constructivist approaches [187], rather than passively consuming static information.

Open-source LMS like ILIAS or Moodle additionally support community engagement by offering interfaces for customization and individual plug-ins. Institutions can customize their learning management systems to their local needs and extend their functionalities. But despite the possibility for massive customization, usually not even the default repertoire of functionality is used in daily teaching. LMSs are usually designed to enrich courses with functionality which can be obtained in front of a PC or laptop, at home or in a library. When considering the evolution of mobile devices, LMSs lag behind the field in of mobile learning.

2.3 Serious Games

Serious Games are games where the primary purpose is more than pure entertainment. They are designed to enrich a serious topic with a full gaming environment. Barbosa and Silva [28] write that “Serious Games aim to add educational or training purposes to traditional video games, in order to take advantage of video games’ inherent engaging capabilities”. Like simulations, serious games can be used to train for situations that are too dangerous or too expensive to teach otherwise [47]. Video games are also able to engage people for hours and days. According to Prensky [168], the estimated amount of time which young people spend playing video games is at 10,000 hours by the time they turn 21. So, video games are very popular and influence people of all ages [101]. Simulations, on the other hand, try to depict a serious environment as realistically as possible [120], in contrast to games which try to be fun. Simulations allow the user to learn the topic in a playful way, but fun is not a main design element. Simulations are not meant to be self-motivating, whereas games should motivate the user solely by the fun it is to play. A well used collaborative simulation in economics is called “*beer game*”. An implementation from Tseng et al. [193] is a good example for a simulation implemented as a specific learning application. The beer game illustrates supply chain effects and lets the participants experience the impact of their actions in an interactive way. Fun and motivation are not excluded, but the main focus lies on the live experience with the economic effects. As the simulation is performed with trainers in a dedicated environment, it does not need to be self-motivating.

In contrast to specially implemented learning simulations, regular games (so called COTSs) can be used in an learning environment, too. For about 40 years, the computer game industry mastered to optimize the gaming experience. It tries to intrinsically motivate the players and fascinate them again and again. Cheng and Chen [53] divide this motivation into two factors: *individual*, which refers to challenge, curiosity, control and fantasy; and *interpersonal*, which refers to cooperation, competition, and recognition.

They claim that the more factors are fulfilled, the higher is the intrinsic motivation. Moshirnia and Israel [151] used achievements with a modified version of *Civilization IV* to learn history of the American civil war. Regular games usually have a multi-million dollar budget and are especially designed to motivate. Placing knowledge within them is a practicable way to make use of their attractiveness.

When regarding serious games in education, their efficient usage has been a controversial topic. Whereas Prensky [168] claims that they are an engaging medium with high potential, Chen et al. [52] notes that learning games were initially “considered learning absorbency killers with no cognitive value”. But with the time, serious games were successfully used in a variety of applications, where learning became one of the largest areas [31, 168]. Interest grows since 2002 to include video games into lectures for educational purposes [73]. But when using video games in a classroom, they have to meet the course’s subject and add value with a reasonable expense [167]. As a result, more and more games were especially designed for educational purposes. But Wallner and Kriglstein [203] state that although they have a high potential to motivate learners as they get them to spend more time with a learning issue, they won’t be able to solve all issues of education. They claim that video games can create a high learning motivation compared to learning in traditional ways. Cheng and Chen [53] state that this success is achieved by the opportunity to actively participate in the events. The player is not just a passive consumer like in classic lectures. Students play a role and slip into the center of attention, away from their day-to-day life with a clear goal.

By now, tons of serious and educational games have been developed and published. One of the first games used for educational purposes was “Where in the World Is Carmen Sandiego?”TM which was published in 1985. As it was used in many classrooms to teach geography and reference skills, it was also the subject of educational research [100]. Another quite popular serious game is “America’s Army” developed by the army of the USA [212]. Besides virtual training for soldiers, its goal was advertisement and recruitment of new soldiers.

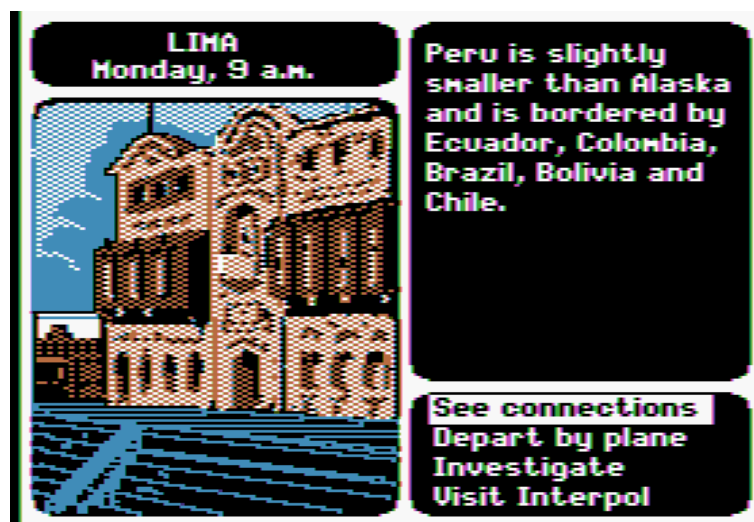


FIGURE 2.1: Screenshot of the game “Where in the world is Carmen Sandiego?”TM

Other Games were developed with a specific learning topic in mind. “Oxyblood” [28] teaches the human blood circulation system. The goal in the game is to successfully carry oxygen through the human body. The players control a small spaceship and have

to fulfill several tasks like protecting red blood cells from bacteria. “Super Energy Apocalypse” [74] tries to teach the players the sustainable use of energy, natural resources and pollution. Specific types of energy produce waste which is then consumed by zombies. The players therefore have to find a balance between the usage of different energy productions and controlling the zombie mob. “RumbleBlocks” [54] focuses on children in a typical kindergarten age of 5-8 years. It is used to teach basic structural stability and balance concepts. The player has to build a tower, stable enough to bear the aliens’ mothership during an earthquake. Another form of game was developed by Kelleher and Pausch [118], who implemented a storytelling game to inspire the interest of middle school girls’ in programming. The Alice programming environment was used to create computer-animated movies. The goal of their project was to engage students of both genders alike, with a “The Sims”-like game environment.

A problem of these serious games is the high specialization to one specific topic. A huge amount of resources are spent to develop a reasonable game for only one purpose. One solution is to provide a basic gameplay functionality which is independent of the actual content. The content can then be created using procedural generation as described by Hendriks et al. [105], or by allowing the user to create his own content. Mehm [145] tried to offer a higher flexibility for the basic game and developed an authoring tool which enables lecturers to create a complete story-based game without the need to write source code. The resulting game is unique in terms of story and content. Lecturers are able to create individual games according to their current courses. Torrente et al. [190] created a tool for custom-made serious games, too: *e-Adventure* [190] offered a graphical editor which enabled the user to create arbitrary adventure games with 3D graphics. The users did not have to actually program the game. That gave them a great level of freedom, but still left them with the complexity of creating a good story, assets and visual scripting. Other serious games limit the overall gaming scenario and only leave the content to be specified by the user. *The Gopher Game* [44] and *WeQuest* [139] use location-based scenarios which can be created by the users. Authoring editors allow the lecturers to create their own location-based game and fill them with their specific content.

Realizing this drawback of a high effort in creating games for very small amounts of users, Mildner et al. [146] developed a basic game, which is fully independent of a specific topic. Every lecturer can feed the game with his or her own questions and answers, which then are integrated into a multiplayer first-person shooter. “Word Domination” combines an engaging multiplayer action game with knowledge transfer in form of quizzes. Students play against each other, as they are confronted with questions about the current course content. Players who perform better in answering the question gain a gaming benefit above their opponents. In another approach, Mildner et al. [147] extended this idea to a Tower-Defense [147] game, which faces the students with several mini-games. These mini-games hold the actual course-related content and can easily be exchanged. Thus, it is quite easy to import new content into the game. The game itself is only a frame to provide fun and interactivity.

Unfortunately, some educational games focus on the integration of correct learning content and decrease the fun [146]. Game developers usually lack the necessary and complex domain-specific knowledge, whereas the knowledge specialists do not have the required implementation skills [73]. To overcome this obstacle game developers have to work more closely with domain experts. Educational games should allow a nearly ideal form of active, self-driven, constructive, motivated, situated, and social learning. They should

not lead to a waste of time, social isolation, a lack of motion, and the danger of addiction [167]. Similar to other forms of learning, not all people are equally motivated by video games [101] or equally triggered to learn with them [58].

In summary, serious games are fully specialized applications, which mostly support only a few different teaching scenarios. They are quite complex in their development and usage. Additionally, developing a good game requires efforts in story writing and level design, which should not be underestimated. Serious games can offer highly interactive learning scenarios with a motivating component where learning takes place along the way of playing the game, but they are also more difficult to use in a course as they do not belong to lecturers' typical repertoire and have to fit well to the current course content. They often have time restraints and a certain necessity for hardware. Additionally, a serious game needs to fit the teaching style of the lecturer and the learning style of the students.

2.4 Games with a Purpose

Another type of games with a serious intention, but without a learning value for the players, are *games with a purpose*. These games should be as much fun to play in leisure time as usual COTSs. But by playing a game, the players generate a benefit for the game host without recognizing it. The players do actual labor for the game's operator, without perceiving it as struggle. These games usually target tasks, which are difficult to accomplish by computer intelligence, but easy to do for the human brain, and wrap these tasks into a fun game. They are used in image labeling [198], image search [201], object recognition in images [200] and task which need good skills in 3D computation [122].

A famous example is "Foldit" by Khatib et al. [122], which gives the players the task to optimally fold a complex protein. The structure of proteins in the human DNA is well known, but the shape in which it is folded is difficult to compute. But as the proteins appear in a shape of an energy minimum, and the energy level of a protein can easily be calculated, the players are free to fold the proteins to their best guess. Afterwards, the best solution of the players' inputs is chosen as solution. This "human computing" often outperforms computer generated solutions.

Von Ahn and Dabbish [199] give recommendations for creating such games which can be used for image tasks. They compare the results of two players with the same task and match similarities. E. g., by looking for matches between the players' descriptions of images, they were able to identify good labels for this image to optimize an image search engine [201]. In another example, humans had to identify objects within an image, and therefore helped to train a object recognition algorithm [200].

2.5 Gamification

In contrast to the serious games or games with a purpose of the previous sections, gamification addresses individual gaming elements which are used in other contexts. Deterding et al. [70] define gamification as: "the use of game design elements in a non-game context". Although gamification is inspired by video and computer games, it uses

only a set of single game elements in a new context to improve the experience of the users. Gamification can further be distinguished from *playfulness* as gamification has a deeper goal. Whereas playfulness means that the main reason for the activity is fun, the focus of gamification lies in a separate purpose. Gamification only improves the user’s experience in doing something for this other purpose. Figure 2.2 shows how gamification can be delimited against games, toys and playfulness. Whereas *playing* denotes a more expressive, free-form, and improvisational form of behavior, *gaming* is structured with rules and a competitive strife towards goals Deterding et al. [70].

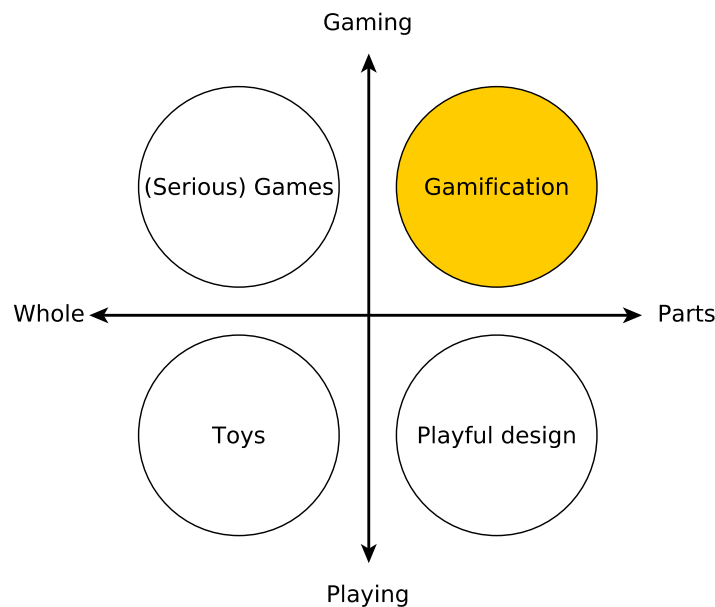


FIGURE 2.2: Gamification between game and play according to Deterding et al. [70].

Computer games are quite popular among typical learners as they are able to motivate them over a long time period. One interesting aspect according to Gee [94] is that computer games have to be learned as well. When combined with real-world topic, gamers engage themselves not only with the game’s ordinary content, but with issues around the main plot. “This is education at its best, and it is happening at home” [94]. The author refers to an example of playing *Age of Mythology* in school, a strategy game settled in ancient Greek. The students informed themselves about Greece gods and legends even after school. Gee states that “researchers have to learn about learning from good computer and video games”, as they use learning principles which were supported by contemporary research in cognitive science (see Bruer [38], Clark [56], Lave [134]). The most important factor therefore is motivation. Motivation drives learning, it “is a learner’s willingness to make an extended commitment to engage in a new area of learning” [72].

Games encourage this motivation by following a set of good learning principles. E.g., games give necessary information at the right moment. People are not very good in remembering and understanding information they have received in the wrong context [37], therefore games usually offer pieces of new information before the player can actually use them. They also confront players with an increasing difficulty, which are challenging but

always do-able, while lectures often operate at the level of the lowest denominator [72]. Thus, games keep their players within the flow [64, 158], as described in Section 2.1.2. Furthermore, games enable the players to not only consume information but produce something on their own [36]. Players create elements in the gaming world as they build their own military base or a whole empire. Being a producer gives a far better identification and long-term motivation with a given topic than only consuming pre-defined elements.

Charsky [49] takes up these principles and lists further game characteristics which motivate and excite gamers. His first characteristic is *competition and goals*. Some games allow players to compete against each other or against the computer, and they confront players with goals which can be achieved [24, 25]. In chess they have to capture the opponent's king, in poker they want to win the money. In a good game, these goals increase in difficulty and complexity, and taking the learning flow of the player into account. An early goal within a strategy game may be the capture of a small outpost, whereas when the player's capabilities increase, he or she may decide to conquer the rest of the gaming world, too. The second characteristic is *rules* [49]. Rules are typically fixed and limit the range of actions of the player. They are important as they can be set to represent reality [24, 25]. Playing an omnipotent god without limitations won't be challenging in the long term. But rules do not have to specify every single bit of the gaming experience. When playing a city simulator like *SimCity*, rules only define the underlying logic of the game. It is almost impossible to build the exact same city twice, as small events within the gaming rules lead to a different outcome every time. Therefore, rules can be understood as the amount of functionality and control, which is given to the players [102, 140]. Another game characteristic is *choice*. Charsky [49] further separates it in expressive, strategic and tactical choice. "Expressive choice" has little effect on learning the game, but it is able to improve the player's motivation. This may be applied towards gamers upgrading the visuals of their avatar, battleship or city. When combined with other game characteristics, expressive choices may increase the immersive experience of the players. This improved immersion should generate a level of motivation which is ideal for higher order learning [71]. "Strategic choice" affects the manner in which a game is played. These refer to the gamer's ability to configure game attributes like difficulty or time constraints. "Tactical choice" represents the way players make decisions about how they play the actual game. It presents the possibilities they have to choose an option 'A' over 'B', the consequences for the further game process, and access to some kind of help [102]. E. g., in a strategy game, after the player made the strategic decision to play the Romans in a simple difficulty level, the way actual warfare is conducted is driven by a series of tactical choices. The last motivating game characteristic listed by Charsky is *fantasy*. Most computer games contain fantasy elements in an attempt to provide exciting and motivating game play [62, 163].

Another characteristic not mentioned yet is the actual presentation to the user. Whereas the characteristics above are meant to give the game a structure and deeper motivation, the first barrier of every software application is the presentation to the user. According to Davis [66], the motivation of the user to apply technology is influenced by its appearance. Therefore, the graphical interface and sound effects can be listed as gamification characteristics as well.

2.6 Mobile Learning

Today’s digital native students are used to carry one or more Internet-enabled devices with them all the time. Studies from Corbeil and Valdes-Corbeil [59] and Ebner and Schiefner [80] show that students usually own many different mobile devices (see Figure 2.3), like smartphones, iPads, Tablet-PCs, E-Book readers and laptops. Therefore, it was only a matter of time until these devices were used in learning and teaching, too. This field is called *Mobile Learning*. According to Traxler [191] and used by Dabbagh et al. [65], *mobile learning* or *M-learning* is defined as learning delivered primarily by mobile and hand-held technologies. Castillo and Ayala [45] define mobile learning as “... the use of mobile devices, such as PDAs or cellular phones in learning activities anywhere and anytime, bringing information and knowledge to situations and places where learning activities take place.”

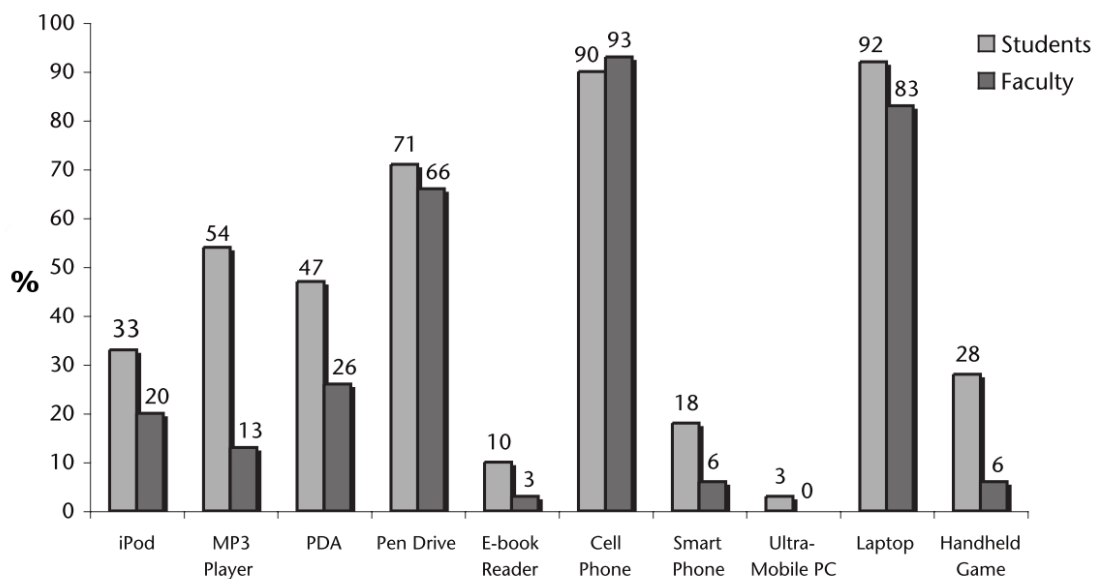


FIGURE 2.3: Distribution of mobile learning devices among students according to Corbeil and Valdes-Corbeil [59] in 2006.

Many lecturers take advantage of the possibility to apply student devices in new mobile learning and teaching scenarios. Such scenarios could be created in which the learning could be taken directly to the real-world objects of interest, without the expense of human tutors [109]. These scenarios face the growing requirements for obtaining relevant information, communication and collaboration at an appropriate time and place [142]. Hwang et al. [109] showed that this not only improves the students’ learning performance but it also lowers their cognitive load. Mobile learning therefore helps to lower existing learning obstacles which were caused by unrelated learning content. Brown et al. [37] state that learning which is isolated from an authentic context limits the construction of new knowledge. Thus, mobile learning can overcome this situation by supporting the students to learn in authentic environments within the real context. Their mobile devices allow them to access the information at the right place and time [109].

Pea and Maldonado [165] state that mobile devices have seven notable features – beside affordability – that influence their rising usage in education. Among these are the size and portability of the devices, the screen size, the communication ability, the range of

available multi-purpose applications and the stylus-driven interface. When these features are favorably balanced, the devices extend the way how students learn more effectively, as they are able to access information from nearly everywhere the whole day long. Today's students, as they are digital natives, expect an integration of mobile technologies into their lives. This behavior should be used to take advantage, by supporting continuous and situated learning in order to improve their overall learning experience [142], particularly, as mobile learning on the students' devices is quite inexpensive compared to PCs.

One of the hottest topics in mobile learning is the usage of Tablet PCs, mostly Apple iPads, in lecture environments. For example, studies by Khaddage and Zeidan [121], Wegener et al. [207], [206] Kugler et al. [128] and Christel et al. [55] investigated the use of iPads in classroom environments. The iPads are used for activation within the classroom and to support self-learning outside of lectures. The results showed that students appreciated the usage during the lectures. But when preparing or revising the content afterwards, they prefer to work with their laptops, as they allow to perform more efficiently [207]. Also, whereas students easily adapt to iPad teaching scenarios during the lecture, they chose the ones for self-learning at home. When contributing their own time (outside of a lecture), they only choose learning applications from which they expect an appropriate knowledge benefit.

2.7 Audience Response Systems

A minimal yet quite prominent solution to increase activity within the classroom with mobile technologies are *Audience Response Systems* (ARS). ARS, also called Clickers [42], Voting Machines [173], Wireless Keypad Response Systems [41], Classroom Communication Systems [77], Electronic Response Systems [115], Classroom Response Systems (CRS) [21], Personal Response Systems (PRS) [129], and Student Response Systems (SRS) [144], typically use hand-held pocket-size devices that transmit the participants' responses to the instructor instantly during the lecture [161]. The scenario is similar to the well-known TV show "Who Wants to Be a Millionaire?", where the questioned candidate can ask the audience for the answer. The audience answers a visualized single-choice question, as shown in Figure 2.4, via small hand-held or build-in feedback devices. The audience's results are then shown to the candidate and can be used for his or her answer.

Similar to that, audience feedback can be used in university lectures. Almost any lecturer knows the situation when he or she asks a question to students. It doesn't matter if it is a knowledge question on the current syllabus or a request for an opinion about the current topic. Students mostly give puzzled looks, and only a minority of them participates actively. But even if puzzled looks can be a useful feedback to see whether the presented material was understood, the students' interaction and engagement are important parts to create a good classroom environment [144]. With a technical ARS, the lecturer can reach the majority of locally (or even remotely) present students to participate in the answering. The systems can be used to anonymously test the students' understanding of a learning unit [119, 129, 153], for timed responses for the fastest right answer [153], exam preparation [129], asking for their opinion concerning course content or course design [205], giving feedback to presentations [153], used as a trigger to start interaction with peers [153], preliminary data collection in preparation of a study [153], for asking controversial questions to start a discussion [153], collect votes after a



FIGURE 2.4: Famous example of a visualized question from the TV-show "Who Wants to Be a Millionaire?".

debate [76], conduct or illustrate experiments on human responses [76, 153], or to simply increase fun and engagement [42, 153]. Similar to the TV show mentioned above, the results are collected, aggregated and displayed to the audience [22]. Usually, the results are directly used to find out whether the content was understood, or rather what the audience's consent is [22, 164]. This immediate awareness influences the instruction, as lecturers can directly adapt their explanations to the students' real needs [161]. The answers can then be a starting point for further explanations or face-to-face discussions [211].

Many studies have investigated the effects on activity, motivation and learning when ARSs were used in lectures. And quite a few surveys already give an overview of the results (e.g., Caldwell [42], Fies and Marshall [87], Keough [119]). Students think that their learning success in courses with ARSs is higher [82, 172, 194], courses are rated less boring [192] and the students' motivation is increased [82]. The usage of ARSs therefore leads to a significantly higher learning success [51]. ARSs can be used to overcome problems with time constraints and large numbers of students as these can hinder the usage of pedagogically beneficial methods like group work, feedback sessions or plenum discussions. ARSs are especially valuable for introducing and monitoring peer learning methods in large lectures [42]. Furthermore, they are useful to activate listeners in courses at unfortunate times, like early in the morning or late in the evening, when students and lecturers are tired [161]. The typical human attention span is no longer than 20 minutes, so recall of information usually drops fast after that [40]. Students say themselves that they get uncomfortable to endure after 20-30 minutes of uninterrupted lecturing [138]. As an ARS interrupts this monotony, it reactivates students, enables collaborative learning and increases their performance [161]. Students are more motivated to engage themselves, so that the overall classroom engagement improves [22, 161, 164]. McLoone et al. [144] even claim that the use of ARSs can increase the students' interaction and attendance, create a comfortable learning environment and raise student satisfaction. Oigara and Keengwe [161] also state, that due to the active learning discussion, students become more critical thinkers. Another main advantage of technical ARSs is anonymity. It is one of the main reasons of its success, as the

use of answer devices lets even reserved students express their knowledge or opinion anonymously. They do not have to fear to be exposed in front of the whole audience [22, 161]. Students appreciate this, and research states that it is one of the main reasons for the increased interaction and activity [21]. This safe environment allows unaltered student responses as the “lack of privacy during voting may prevent completely honest votes” [42]. These honest responses then give a realistic impression of the students’ true state of knowledge and their knowledge gaps. Lecturers are therefore able to specifically work on the revealed gaps. As the results in Section 5.2.2 will show, lecturers can be shocked, when realizing how many students did not understand the material. With a true insight in this knowledge, the “house of cards effect” when students understand new content poorly because it is based on other poorly understood content [42], can be avoided. In comparison to classic question and answer situations, this can be achieved for almost all students of the course, as everyone has the chance to answer for himself [119], not just the few that would answer classic oral questions. In addition, the aggregated results have another benefit for students. Perry et al. [166] state that students have expressed relief and an increasing confidence level when realizing that they were not the only ones struggling to understand the current lecture material. So, ARSs can offer time-effective alternatives to traditional learning methods. They can make classes more learner-, knowledge- and assessment-oriented [164]. Therefore, ARSs can be highly recommended for higher education teaching.

The early Audience Response Systems were usually called clickers or voting machines, as they typically consist of a hardware device, as shown in Figure 2.5 per student and a central station which collects the responses. Some systems had build-in devices, firmly integrated into the classrooms furnishings whereas others used hand-sized devices ([77]) which were distributed and recollected. The main goal of early clicker systems like Classtalk [77] was to improve the involvement of every single student with the help of technology. The teacher transferred three to four Classtalk tasks per lesson to the students’ devices, which were calculators, organizers, or PCs at that time. ConcertStudeo later used an electronic blackboard combined with hand-held devices and offered exercises and interactions such as multiple-choice quizzes, brainstorming sessions, or queries [67]. Murphy et al. [154] and Kay and LeSage [116] discuss advantages and disadvantages of commercial ARSs which usually sell special hardware clickers together with a main station. But the authors pointed out that the purchase of hardware devices may cause a lot of additional overhead, like securing devices against theft, updating, handing out and collecting them, providing of batteries, handling of broken devices, and instructing teachers as well as students about their usage. Other disadvantages are higher prices of dedicated devices as well as their coarse handling by students. And despite the high mobility as some of these systems were very portable and fit into a suitcase, they still had to be transported to the classroom.

Scheele et al. [177] therefore developed the Wireless Interactive Learning (WIL/MA) system to support interactive lectures [126, 177]. It consisted of a central server and a client software part, too. But the software part could easily be multiplied on commercial PDAs. The components communicated using a WiFi network specifically set up for this purpose. The software consisted of a quiz, a chat, a feedback, and a call-in module and was designed to be easily expandable. The main problem was that students needed to have a JAVA-compatible hand-held device, and they needed to install the client software before they could use the system.



FIGURE 2.5: Picture of a hardware clicker device.

(Source: Editallthearticles

https://de.wikipedia.org/wiki/Audience_Response_System)

Although hardware clickers usually had high costs (US 20 – 60 per unit [161]), they were quite popular due to easy installation processes and their ease of use [211]. Nevertheless, these systems were almost always closed systems. As a consequence, purchasing a system binds an institution to a particular provider or manufacturer [211]. This leads to decreased flexibility and can discourage universities from buying such systems. Furthermore, due to their specialized hardware, clickers have limited input capabilities which mostly is limited to single and multiple choice questions. And although they are portable, they require the instructor to procure a large number of devices and distribute them [144]. Abramson et al. [21] state that one of the main disadvantages that lecturers mention is this initial time investment which is associated with their adoption. As the clicker hardware is a very tangible piece of material, it is mainly focused on local usage. Clickers are quite useless for distributed classrooms, as they typically use short-range communication like infrared [144]. Some systems additionally worked very restrictively, for example they did not allow fully anonymous answers as a possibility to ask clarification questions [161].

With the advancement of mobile technologies and hand-held devices, more and more applications were developed and deployed which worked directly on the students' mobile devices [21, 22]. Most of them follow a similar architectural structure: a dedicated interface to create the content and an application for the students' devices which enables the response to the central interface. Applications like, *Clicker* [170], *Click2Vote* [211] or *QuizIt* [22] are only a few examples of the bring-your-own-device era. Abramson et al. showed in 2013 that "86% of undergraduate students at American colleges and universities use laptop computers, while 62% of those students use a smartphone" [21]. Therefore, making use of this trend was a logical and economic next step. By now, the students' mobile devices evolved in technical functionality and propagation, they not

only allow an easy interaction with the device, but also support the playback of multimedia content like audio, text, images, or videos. This new generation of lightweight ARSs could also be used more easily in combination with different learning materials like lecture recordings and e-books [197]. As students now carry their voting devices anyway and modern ARSs usually communicate via the Internet, it became easy to enlarge the classroom virtually and decouple feedback from time and location. Students can participate remotely in virtual lectures or repeat quizzes for themselves when watching a video recording of the lecture. Technically, it is only a slight difference if students join via the local university WiFi or through the Internet from the other side of the world. Some of the new ARS applications could also be integrated directly into the institution's existing infrastructure like the learning management system (e.g., Zender et al. [211]). Therefore, also for the lecturer side, no additional software or devices were necessary.

When it got quite easy to develop Web-based ARSs, a ton of new systems appeared. Some systems only needed a Web server where they are hosted, so that they were fast to create and cheap to maintain. Kundisch et al. [130] counted a nearly even distribution of commercial and free products. Examples of commercial products they listed were *ActiVote*, *ActivEngage*, *eClicker*, *eduVote*, *GoSoapBox*, *learning catalycs*, *Mentimeter*, *Let's Feedback*, *Poll Everywhere*, *QuestionPress*, *Softclick*, and *i>clicker*. Examples of free software listed by Kundisch et al. [130] were *ARSnova*, *Socrative*, *Kahoot!*, *inVote* and *PINGO* (also see [129]). Because developing new applications only needs a short period of time, there is an additional number of local university solutions like *SMILE* [205], *BeA* [135], *AuResS* [111], the TUL system [110] or *BrainGame* [188].

Although the majority of studies state mainly the benefits of using ARS applications, some studies did not recognize significant differences in learners performance (e.g., [164]). Many studies also mention the drawbacks and difficulties when using ARS applications. For example, it needs high efforts to formulate good questions and to integrate the ARSs usefully into the lecture [129, 159]. The ARS should be used throughout the semester, and has to be taken as a serious part by the lecturer. Students have to recognize that ARS are a fixed part of teaching, as the lecturer takes it seriously and reacts on their current inputs [159]. The individual ARS sessions should further be timely relevant for the actual course material [32]. Also, the commitment in time for preparing, performing and discussing the student feedback should not be underestimated [129, 159, 181]. When students are asked about the weaknesses of ARSs, they mention that it is most important for them to work correctly. Technical difficulties are the most disturbing aspect [119]. A few students also state, that using an ARS disturbs the lecture as it creates noise and motion in the classroom. The distraction by using student devices within the classroom is a fear also shared with the instructors. "While Internet-connected computers can be invaluable for research and learning, students often use them for other purposes while in class. There is considerable evidence that students who are permitted to use their laptops for search, social networking, or self-directed browsing suffer significant reductions in learning compared to peers who do not use laptops" [21]. As a consequence, mobile devices had been banned from some classrooms [21]. In contrast, Kim et al. [123] conclude from their study that using smartphones for a learning in-class purpose (in this case a Twitter-based scenario) prevents students from using them for non-learning purposes during the time of the lecture.

There are still some technical drawbacks with mobile devices in ARS, even though these limitations may only be of temporary concern regarding the rapid evolution of technologies [162]. But, these limitations exist and need attention. According to Park

[162], mobile devices have limitations in limited physical attributes, software application, network speed and network reliability. Physical attributes are screen size, weight, insufficient memory, and a short battery life. Limitations in software applications are a possible lack of built-in functions, difficulties to add further applications, and the effort to learn how to handle the devices properly. Marzouki et al. [142] mention additional social and pedagogical issues. There are accessibility and cost barriers, frequent device changes, steady technology advancements and a rather slow development of appropriate learning theories for the mobile age.

Recent research on audience response systems mostly focuses on improving the flexibility, expandability and interoperability. Web-based systems should have improved user interfaces that allow lecturers to add new questions on the fly, to create a collection of questions, or to check answers immediately [110, 111, 135, 188]. The lecturer may also export the collected data for later analysis. Other functions include the possibility for students to update their votes, or to support user authentication for enabling user specific grading. Because of the huge variety of systems, approaches like *Click2Vote* [211] focus on the interchangeability of questions and results through the applications which lecturers typically use.

Beside high asset costs for commercial solutions, many ARSs are functionally restricted to multiple choice, numeric and similar answer formats. The didactic benefit is therefore limited to the amount of realizable scenarios. When designing a course, the lecturer has to search for the tools and methods he or she requires to realize the teaching goals. At this point, lecturers often face the problem that there is no tool fitting exactly their needs and expectations. In the end, they have to arrange themselves with semi-ideal solutions. In other cases, lecturers use the tools for the sake of the tools, not taking into account that the tools have to be subordinated to their teaching goals. Reasonable course planning therefore needs flexible instrumentation that actually fits the stated goals.

2.8 Generic Approaches for E-learning Systems

One of the main disadvantage of audience response systems discussed in above, is the high effort of adding new or modifying existing functionalities. Usually, every new kind of learning object, e. g., a new question type, needs an additional implementation. Every variation of a lecturer's teaching scenario has to be designed and implemented separately.

Therefore, other systems were developed to offer more flexibility and extensibility. These systems tried to offer more task and answer formats and to support different kinds of individual feedback. *MyMathLab* is one prominent example that was evaluated as highly useful by Chabi and Ibrahim [46] to improve the math skills and learning success of the students. The system analyses the learning progress of students and generates equivalent tasks.

Another example comes from Seemann [179] who created an individual feedback and assessment tool that 'understands' mathematical expressions. It allowed teachers to create exercises with a wider range of question types than many other learning management systems, which were often restricted to multiple choice questions.

Other fields of e-learning are far more focused on general valid, generic data models. In learning analytics, as will be described in 2.10, generic models like the MOOCdb by Veeramachaneni and Dernoncourt [196] were quite popular. The MOOCdb uses a generic database schema to store many different types of learning content in a reasonable number of database tables. The benefits are that analyses can compare different types of learning content more easily, and new types can be added without changing the database structure.

Consequently, we also wanted to extend the flexibility, adaptivity and extensibility of our system and offer a highly generic software tool. Instead of focusing on math skills, our software tool is applicable to a wide range of subjects, it is easily extensible and can be fully customized to the lectures learning scenario.

2.9 Authoring Tools

It causes an initial effort to create new content in a given framework or environment. The specifics and language of the framework have to be understood and mastered. Graphical authoring tools separate the users from the low-level mechanics and languages to save them from learning a new programming or definition language. One of the first graphical authoring tools was introduced by Apple Computer, Inc. for the Macintosh computers in 1987 [180]. It was called *HyperCard* and was an object-oriented, graphic programming language. Sholl et al. [180] used it to create rapid prototypes of CASE interfaces very efficiently. They concluded that it offered a significant reduction in the development time and assumed that this will further improve when HyperCard would be upgraded further. HyperCard therefore stands as one of the first commercial authoring tools for complex, hypertext-based content. With the technological evolution, authoring tools became more popular and were eventually used in learning technologies. Within this environment, authoring tools are usually understood as software applications which support teachers and instructors in designing and creating e-learning products. Dağ et al. [68] give a more specific explanation. According to them, authoring tools are systems that to provide features for users to assemble abstract objects, such as hypertext or multimedia content. By sorting and linking these objects, authors are able to produce advanced learning content, such as simulations, instructions, or quizzes [68].

Besides the benefits of these authoring applications, creating sophisticated e-learning resources remains a difficult and complex task, which requires technical and didactic skills [98]. Especially the creation of modern, rich learning experiences is quite challenging, as the integration of multimedia content and the enabling of interactivity usually require the technical understanding of programming structures and algorithms. According to Murray et al. [157], these challenges can lead to high asset costs and large technical overheads. This slows down the propagation of new technologies in higher education environments. The use of authoring tools tries to address these problems. Authoring tools try to provide a graphical user interface which isolates the user from the technical core of the system. The users therefore can focus on the didactic content, as the technical requirements are reduced. The process for preparation is more structured and efforts in time and money can be conserved [156].

This fact caused the development and usage of a large variety of authoring tools for the e-learning sector. But a universal categorization for these authoring tools is difficult to

make [156]. They range from learning content management systems (LCMS), intelligent tutoring systems (ITS) [156] to advanced learning environments (ATLE) [155]. The existing authoring tools vary widely in their amount of capabilities and features. Early systems, like *Macromedia Authorware*, were built as local applications. But the need for web-based, distributed applications grew since the start of the Internet [185]. These early systems usually failed in the interoperability and reusability of their learning resources. But lecturers need to share and distribute their content easily among several systems. Baumgartner et al. [29] addressed the didactic relevance of this interoperability as a fundamental step in the evolution of authoring tools. They pointed out, that this was a significant improvement for lecturers who create and distribute rich e-learning resources. Based on that observation, learning technology standards like SCORM and LOM have been developed and were used by many authoring tools [29, 97]. With these standards, a focus was set on *learning objects*. Learning objects are small, meaningful and enclosed learning units which could be reused in various contexts, shared or combined to generate more complex learning objects [93]. An important feature for an authoring tool therefore is to support these learning objects within a common e-learning standard. Applications like *HyCo* [93] focused on these learning objects. *HyCo* is a hypertext authoring tool, developed for semantic learning objects. Later approaches from Gallego et al. [91] used context awareness to offer additional recommendations for these semantic learning objects. The user's activities were analyzed, together with the educational context, to offer these recommendations. Others like Gordillo et al. [97] and Johnson et al. [113] focused on creating reusable quiz objects, while regarding learning object specifications. Beside these specific use cases, Muñoz-Merino et al. [152] investigated the interoperability between authoring systems and learning management systems. They discussed current learning technology standards and presented an XML-based authoring tool solution.

Murray [156] discussed the trade-off between usability and power of authoring tools. The goal was to create e-learning content simpler, faster and more efficiently, without losing flexibility. But this trade-off between flexibility and ease of use seemed to be a large limitation of existing authoring tools [156]. Sayago et al. [176] also criticized the high affinity of many authoring tools to the technology standards which decrease the overall usability. They tried to overcome this problem with their own application but the prototype showed some outstanding design problems. Further manifestations of these rapid authoring tools have been developed to create learning content in an uncomplicated and more time-efficient way. They were usually easy to operate and offered interfaces to applications already used by lecturers. Yuuichi et al. [210] introduced such an application in 2006. They wrote an lightweight AJAX-based rapid authoring tool for creating classroom quizzes. Its major motivation was to replace the old and complex authoring tools of common learning management systems. The new generations of authoring tools like *Adobe Captivate* and *Articulate Storyline* therefore used special usability and productivity features [195]. In addition to the improved usability, some approaches focused on pedagogical aspects. Capuano et al. [43] proposed a rapid authoring tool which used predefined pedagogical templates.

Nowadays, with the current distribution of mobile devices in lectures, authoring tools have to satisfy the requirements for new teaching methods. Mobile devices are typically used for short-time, in-class audience response scenarios like quizzes [4, 12, 181]. Authoring tools therefore are focused on quiz/content creation and should be aware of reduced screen sizes and limited computational power. They should provide responsive templates and HTML5. Technologies like Flash experience a decreasing support

on mobile devices, due to their ineffective drain of resources and should therefore be avoided.

2.10 Learning Analytics

Learning Analytics is a field combining techniques from several other fields. Among them are *Academic Analytics*, *Educational Data Mining*, *Social Network Analysis* and *Business Intelligence* [78]. One good definition of Learning Analytics is used by Siemens and Baker [182] and comes from the international *Learning Analytics and Knowledge Conference* in 2011: “*Learning analytics is the measurement, collecting, analysis, and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs.*” As Ferguson [85] points out, this definition is usually extended by assumptions like “learning analytics have to be able to use machine-readable and preexisting data and further possesses the techniques to handle numerous datasets”, because the analysis of ‘big data’ cannot be handled manually. But several other definitions can be found in the literature that only differ slightly from the one above. Agudo-Peregrina et al. [23] stress the fact that the data for analyzing purposes should be extracted from a virtual learning environment. However, a tool should be usable in more than one virtual learning environment. Other definitions stress the fact that the evaluation of the data should support the decision making and provide an insight into the learning and teaching methods [189]. Therefore, all of the stakeholders should benefit from learning analytics: Students should be able to evaluate their own strengths and weaknesses more accurately, and teachers should gain a better insight into the state of their class [124].

Dyckhoff et al. [79] summarized the goal of learning analytics as tracking the students’ activities and interactions with the system. Tools have to be used to collect the wanted data from several systems and offer educators and students information about these. The important learning aspects should be highlighted and the overview should enable a comparison between different groups of students. Educators can then discover patterns in their students’ behaviors and should identify possible capability gaps. They should be enabled to validate their used teaching material on its usefulness for students and adjust it accordingly. Students should be able to compare themselves with classmates. This should motivate them and may adjust their learning behaviors to become better learners [79].

Ferguson [85] splits the field into three parts:

- Educational Data Mining to observe the technical challenges,
- Learning Analytics to observe the educational challenges,
- Academic Analytics to observe the political and economical challenges.

But the borders of these parts are smooth. *Educational Data Mining* focuses on the perceptions that can be automated by a system [182], whereas Learning analytics tries to leverage the human judgement and empower the instructors and learners. In contrast, *Academic Analytics* collects the data for educational administrators to benchmark and improve the effectiveness of educational institutions [86].

For implementing a learning analytics system, various frameworks have been developed to consider the environment and the necessary parameters. Greller and Drachsler [99] mention that learning analytics has many critical areas like the ownership of data and the ethical usage. Greller and Drachsler [99] and Drachsler and Greller [75] identified six critical dimensions which had to be considered when performing a learning analytics project. These dimensions were *Stakeholders*, *Objectives*, *Data*, *Instruments*, *External Constraints* and *Internal Limitations*. After the environmental parameters were clarified, Clow [57] presented a learning analytics cycle, which describes the successful process of learning analytics. The four steps of the cycle are *learners* which generate *data*. This is then analyzed with *metrics* and could lead to *interventions* by the lecturer. The learner then creates new data, and the circle is repeated.

Within learning analytics, one of the rising topics focuses on the mobile aspect. Mobile technologies are increasing rapidly and therefore carry the field of mobile learning analytics with them. Mobile learning analytics focuses on learning with and through mobile technology [191]. It focuses on data created by mobile learners on their smart devices [26]. Therefore the mobility of the students and its impact on the learning process is regarded separately [90]. This also considers the social aspects of today's mobility, which is influenced by Facebook and Twitter and has a impact on the students' learning behavior.

Researchers like Macfadyen and Dawson [137] showed a significant correlation between engagement tools and learning outcomes in online courses. A high participation in forum discussions can be correlated to the course's final grade. With these kinds of knowledge, lecturers can engage their students to participate more actively in the classroom. But few of the tools used today correlate the teachers' data to the student data [79]. Therefore, the knowledge of the lecturer's teaching styles is not as large as it could be. But Dyckhoff et al. [79] also point out that the presence of a tool which is able to provide that information, does not necessarily have an impact on the students' or the teachers' performance.

Data Models

Learning analytics applications usually collect data from different sources and in different formats. To be able to analyze and compare them, they have to be stored in a database model which is valid for several types of learning data. Therefore, a remarkable part of the learning analytics community is engaged in developing and evaluating suitable data models.

Lukarov et al. [136] categorized the data models into four categories with six concrete examples. Their models were focused on analyzing motion data created by logging the students' behavior within a system. These were: *Event Centric*, *Activity Centric*, *Object Centric* and *User Centric*. They identified Contextualized Attention Metadata (CAM) [160] and IMS Caliper from IMS Global as Event Centric models. They focused on the monitoring of user interactions with learning environments. The activity centered *Activity Stream* [160] is quite similar but spares to consider the individual sessions. It only regards the activity of a user with a learning object. Object-centric approaches like *NSDL Paradata* [160] only collect aggregated data about resources.

In contrast to the motion data models described by Lukarov et al. [136], Veeramachaneni and DERNONCOURT [196] tried to develop a data schema to store a huge variety of different

data used in learning applications. They presented data standards which were especially suitable for MOOCs. They use a generic database schema to store many different types of learning content in a small number of database tables. The benefits are, that analyses can compare different types of learning content more easily and new types can be added without changing the database structure.

Fortenbacher et al. [89] builds upon this generic model but tried to further simplify it and specialized it to their learning management systems and learning analytics application. The previous application *LEMO* [83] was successfully transformed to the new generic data model, so that the connection to further learning management systems became possible.

In consideration of this simplified, generic data model, Schön et al. [10] also developed a small, generic model to collect the information of the learning activities at the University of Mannheim. The model was similar to the object-centric models of Lukarov et al. [136] and Fortenbacher et al. [89], but the focus was set on the analyses of the content of learning objects, especially student-generated quiz answers.

Learning Analytics Tools

During recent years, several tools have been developed to collect the necessary information and process it in a way that allows analyses of teaching and learning processes. One very large and student-focused project was *Signals* at the University of Purdue. Arnold and Pistilli [27] analyzed the data of more than 11,000 students and applied statistical techniques to predict the students with a high probability to drop behind. The tool provided some intelligence by guiding its users to appropriate recourses and explaining their usage to them. For this purpose, *Signals* collected data from several sources and transformed them into compatible formats. Each student got a personal level of risk value which could also be used by the lecturers to schedule an intervention in the form of an e-mail or a personal meeting.

Another, more exploratory approach was *eLAT* (exploratory Learning Analytics Toolkit) by Dyckhoff et al. [78]. It allows lecturers to explore and correlate learning object usage, user properties, user behavior, and assessment results with the help of graphical indicators. One of its goals was to provide real-time processing of the necessary data within milliseconds to allow individual data analyses for lecturers. They should be able to self-reflect their technology-enhanced teaching and learning scenarios. *eLAT* had a connection to four Virtual Learning Environments, but it offered a fixed set of analyses and would have to be extended for adding further ones.

In contrast to the individual feedback, the *Predictive Analytics Reporting Framework* (PAR) was developed for large-scale analyses in higher education by the Western Interstate commission [202]. PAR is a data-mining project which uses analyses of large scale federal data sets to support the understanding of student dynamics and in particular drop outs. To identify the main drivers that are related to the students' progression, they aggregated data of more than 1,700,000 anonymized records. A further focus of the project was the search of parameters which influence the tendency of online students to stay enrolled. Their results showed that the analyses provided a useful benchmark to explore student drop outs and dynamics at different course levels. As a conclusion, they were able to show that the courses which were attended at the beginning of the college career had a higher influence on the students' dis-enrollment probability.

In the mobile sector, Fulantelli et al. [90] presented *MeLOD* (Mobile Environment for Learning with Linked Open Data). It provided geo-located learning materials which should support the students' informal learning experiences while visiting interesting places. They therefore got additional information on the current location and were able to vote and comment them. This information was then fed into the database together with the social learning activities of the students.

In conclusion we have now presented a view of related research to ARSs and interrelated fields from the early beginnings of e-learning activities to the modern times of smartphones and iPads. Building upon this knowledge, the following chapters will introduce two new models and their implementations. The first model will describe a simple but polished ARS which has the benefits of the ARSs mentioned above, without most of their drawbacks. The second model will build upon that and will describe a far more complex application which is able to support a large variety of different and customizable teaching scenarios, not limited to plain knowledge quizzes.

Chapter 3

Model and Design

3.1 Concept

Nowadays, most students own a smartphone or similar mobile, Web-enabled device like a netbook, laptop, or tablet PC. All these devices are quite powerful and have at least built-in wireless LAN access. The main idea of our concept is to exploit the proliferation of these devices to improve teaching and learning in university lectures individually, customized to the teacher's style.

Our approach is based on the following five steps:

1. Digitization
2. Acceleration
3. Gamification
4. Generalization
5. Customization.

Digitization: The first step is to create a digital representation of the already existing analogue teaching scenarios. Usually, teachers have a set of teaching methods for their everyday work. These methods range from asking simple questions to the audience up to giving complex computing tasks to the students, which should be solved in small groups. The tasks are usually written on a sheet of paper or presented verbally. Therefore, the first step is creating a digital representation of established and proven methods. One of the most popular examples is to digitize quizzes. This digitization directly offers a high benefit as the number of participants does not increase the effort of correction anymore, because this is now done automatically. Furthermore, the amount of work to create a digital quiz is not increased compared to the effort of preparing a test on paper. Of course, certain scenarios are more complex to transform than others. The mentioned tests are quite easy to digitize compared to a complex peer review scenario, where students have to write weekly articles and anonymously review the work of others. Therefore, the basic step is to digitize the teaching scenarios that lecturers already use in an analog form. Presumably these were well proven from the didactic side and have

an equal form or lower amount of effort in preparation, execution, or evaluation when digitized.

Acceleration: In a second step, the process of executing a teaching scenario should be accelerated. The usage of a digitized version should consume significantly less time than the analog alternative. But, it has to be distinguished between time consumed by content and administrative work. The time which is necessary to prepare good and didactic useful content is difficult to influence by technology. Creating good questions and inspiring content requires human creativity. On the other hand, the time which is consumed by setting up a quiz round, collecting the students' input and arrange a discussion can be optimized by technology. Some of these technologies are available in the locally established software systems, the infrastructure of the university campus and the functionalities offered by modern mobile devices. Using these resources reduces the effort of learning new applications in the first place and reduces the time consumed by the lecturer to administer the scenario execution. The local learning management system should therefore be extended with further functionalities. Students and lecturers are already familiar with the LMS and can adapt extensions to the known system more easily than learning a completely new system. The local system should allow lecturers to create, maintain and perform ad-hoc quizzes during a lecture without an additional system. Beside the established software the basic infrastructure should be incorporated, too. Instead of setting up a dedicated wireless network (e.g., Scheele et al. [177]), the university's WiFi and the mobile Internet of the students' mobile phone providers should be exploited. This should significantly decrease the effort to set up a new teaching scenario. Another option to accelerate the scenarios' set up is to remove access barriers. Reducing the efforts to log-in to the current scenario decreases delays and nuisances. Typical barriers are the installation and configuration of special applications on the students' mobile devices, manually entering a URL and a separate authentication via a login screen. To avoid these barriers the application itself could be visualized as a Web application on the students' devices, which needs no further preparation or installation. With the start of a quiz round, a QR code and a link in plain text could be shown on the projection screen. The students could participate in the quiz by scanning the code with a barcode reader, or manually enter the URL in a mobile web browser. After the quiz is finished the aggregated results would be displayed on the same projection screen, and the lecturer would be able to discuss them with the students. Figure 3.1 shows the concept of an application, which already digitized a knowledge quiz for the use of the students' mobile devices and already uses accelerating concepts.

Gamification: After digitizing and accelerating the teaching scenarios, they should become more fun. Adding game elements increases the near-term and long-term motivation even more. Aspects like graphics, sound, interactivity and achievements influence the students' experience with the scenario. Especially in the long run, a steadily high motivation becomes important as the students' first interest decreases. Gamification should help to keep the participation on a high level during the whole semester. Section 3.2 will describe the different gamification aspects in more detail.

Generalization: The next step is to generalize the underlying concepts to support a larger variety of teaching scenarios. Many existing scenarios already use a manageable amount of similar basic concepts. But many of these concepts are highly specialized when

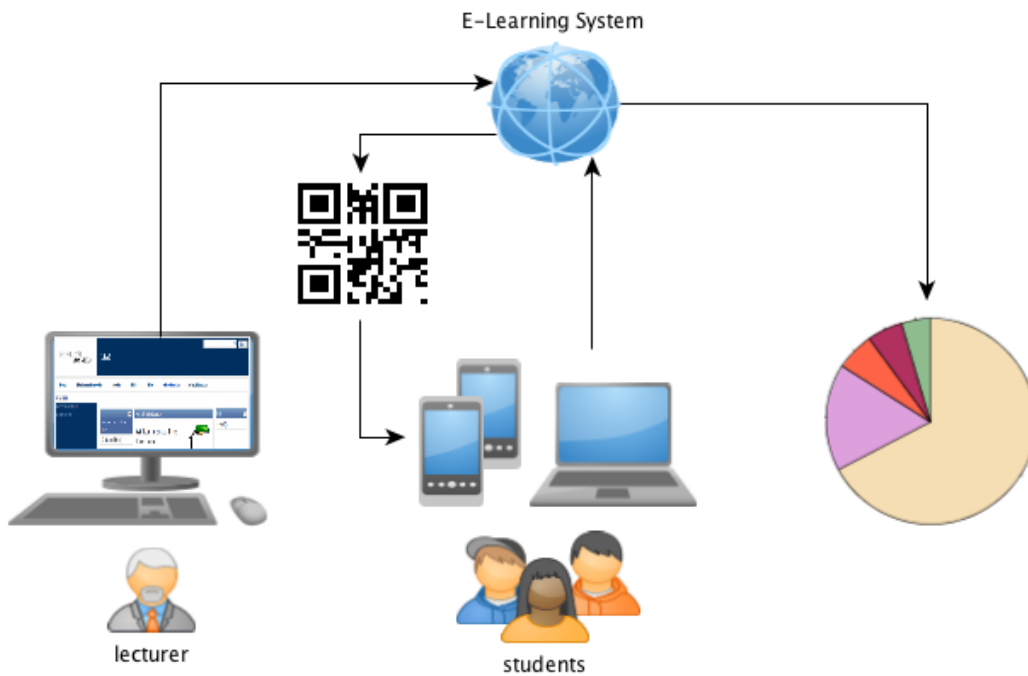


FIGURE 3.1: Concept of a quiz application using students' mobile devices.

the scenario is digitized. They perfectly fit for the planned usage, but cannot be used for further ones. When considering similarities between different scenarios, it should be possible to create a universal model. This new model should then be valid for a higher variety of different scenarios. As an example, a mathematical question with a textual description of the problem and a numerical answer possibility is a quite specialized construction. It can be generalized into a question with different types of answers. The numeric answer would only be one out of several types in the more flexible model. In a more abstract view, it will not be necessary to have a specified implementation for every question type but to look at every question object as a composition of elements which form a question, several answer possibilities and a correct result. In that way, our model would be able to describe a huge variety of different questions with the same structure, and with only one software implementation. In an even more abstract view, questions and answers could both be seen as objects with different sets of attribute relationships. The question may be a question object with a connection to an answer object. Whereas the question may only have one textual attribute which contains the question's textual description, the answer object may also contain an empty attribute for the user input and a hidden attribute for the correct answer. An approach towards such a generalized model is described in Section 3.4.

Customization: After transforming the model into an abstract concept which should be able to describe a huge variety of individual scenarios with different structures and contents, the lecturer should be enabled to create and customize these scenarios for himself. Because every lecturer uses his or her own set of proven teaching methods, it is nearly impossible to describe each of them in advance. The individual requirements are then postulated after the system is introduced, but modifying the application afterwards lead to a high effort which should be avoided. We conducted that the lecturers should be

enabled to create and customize their own scenarios with the help of a construction kit. This allows lecturers to adapt well-established scenarios to their own needs. In many cases the individual requirements differ only slightly from the already existing scenarios, so that modifications should be easily accomplished. In addition to the initial modification, further improvements of the scenarios are reached during teaching. Lecturers can adapt the scenarios to changes in the class size or misestimations in of the students' behaviors. E. g., when a complex calculation scenario overestimates the students' computation skills, a hint functionality could be added to the scenario, so that it will lead to the desired result in future tests.

3.2 Gamification

The variety of methods used to support teaching and learning processes during university lectures is fairly huge. But in our work, the focus will be on how to match the students' real state of knowledge with their own and the lecturer's impression of it. This work mostly uses different kinds of quizzes and similar methods. After the digitization and acceleration of these methods, as described in Section 3.1, the possibilities and varieties grow and open up new fields for further improvements. When adding additional gamification elements, a simple quiz can be transformed into an interactive multimedia experience. Today's quizzes appear in various shapes, from plain paper sheet mock exams up to fully specialized, graphical astonishing serious games. And as described in 2.5, several characteristics exist to gamify a users experience with a software application. But when focusing on quiz-like applications in a classroom environment, not every game characteristic stays suitable.

Designing a challenging scenario which keeps the students in the flow is furthermore a didactic task. Choosing the right content in a fitting difficulty is primarily a task for the lecturer and not directly a task for the quiz application which is used in a limited time-frame during the lecture. But this could be different for long-term quiz applications which were individually processed by students at home in preparation of their exams. The same explanation fits to the characteristic of fantasy. Although fantasy is a motivational factor [62, 163], it has to be accepted by the students. Within this work, it is assumed that accepting fantasy needs time and a fitting environment, which is not necessarily given in a university lecture. Therefore, the used gamification elements focus on the more technical aspects of the application's presentation and interactivity. These gamification features are:

- Digitization
- Interactivity
- Graphics
- Audio
- Multiplayer
- Achievements
- Mobilization.

A plain paper sheet shows no gamification at all, whereas a modern block buster game is fully gamified. Between these extremes are many variations with different gamification profiles. Some applications could be fully developed in a selected set of features yet completely ignore others. The features do not have to be implemented in a strict order; different applications of the same topic may show a completely disjoint set of gamification features.

Digitization

As explained in Section 3.1, digitization is a main step towards gamification. Digitizing scenarios enable a completely new set of further gamification features as they are easier to implement. Facilitating sound and achievements within a software application is cheaper than equipping classrooms with hi-fi equipment and awarding the students with medals. Furthermore, digitizing itself can increase the fun. Right from the beginning students enjoy working with software applications, as they are a welcome diversion to their usual lectures. But this effect not always lasts over time, as students usually become bored very quickly if they aren't motivated by further stimuli. Figure 3.2 shows an example on how basic knowledge quizzes may appear in the gamification spectrum when digitized. The quizzes may be fully digitized, but would have only low ratings in other gamification categories. Whereas graphics might be sufficient, audio might be limited to a single sound when pressing a button. The interactivity could be limited to clicking buttons within the application, and there wouldn't be any communication or comparison to other players. The low value on the achievement scale may be justified by a single message of accomplishment when finishing the test.

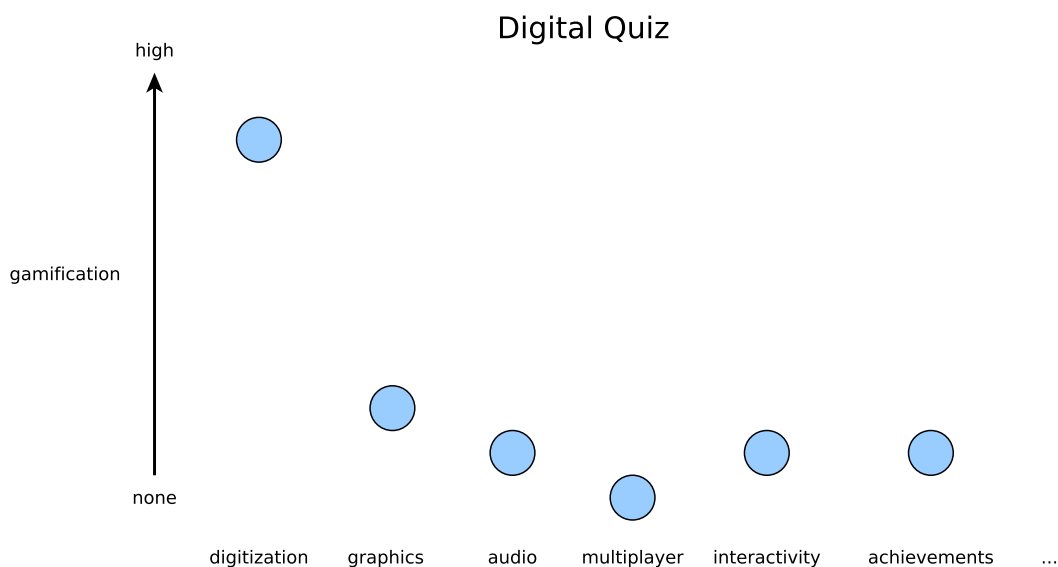


FIGURE 3.2: Low amount of gamification in an Audience Response System despite a high degree of digitization.

Graphics

Graphics are probably the most notable gamification feature. They can range from primitive textual representations of black letters on a white background up to fully colored, highly realistic and animated 3D worlds. An appealing and intensive graphic experience might increase fun and motivation. Independent of a specific game content, just looking at a beautiful graphic may improve the users motivation. A screenshot of the early text adventure game *Zork* (released in November 1980) is visualized in Figure 3.3. *Zork* took place in a fantasy world, but the player had to imagine the entire world for himself, as the whole environment was described with white text on a black screen. In contrast, Figure 3.4 shows a screenshot of the modern fantasy game *Skyrim*. The two games have a comparable world setting, and both use a fantasy background story. But even if the whole player interaction of *Skyrim* was also via text input, experiencing the modern, beautiful graphics usually feels more enjoyable than reading plain text.

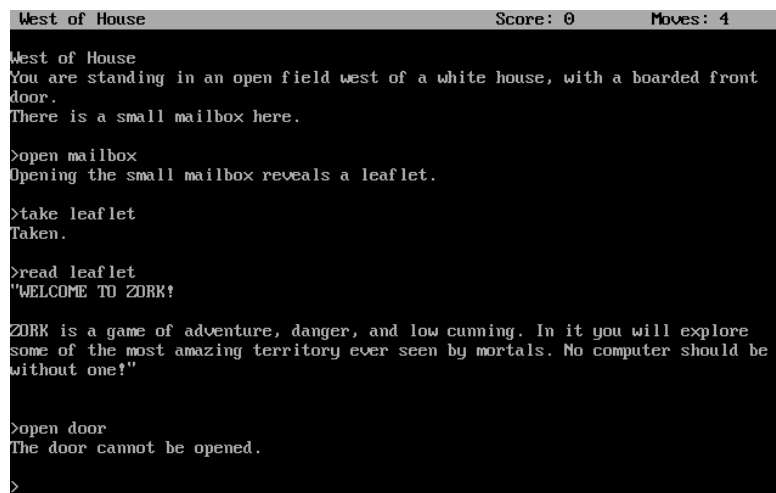


FIGURE 3.3: Screenshot of the game *Zork*.

(Source:

<http://www.heise.de/imagen/swvz-programme/Nx0Gj1KNK3VoYQ40z9hbCejGLTo/content/Zork.png>)



FIGURE 3.4: Screenshot of the game *Skyrim*.

(Source:

<http://www.elderscrolls.com/assets/files/tes/screenshots/Khajiitmalewlegal1.jpg>)

Audio

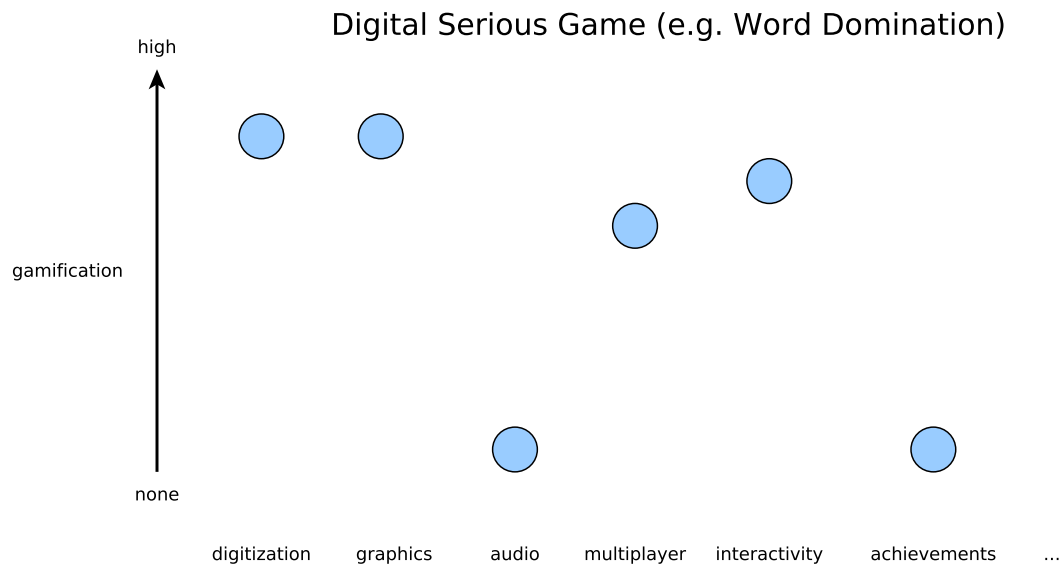
Besides graphics, audio is another important parameter to increase the gaming experience. It ranges from long and unnoticed background melodies to short effect sounds. Endless, encouraging background tones will motivate the user during playing, without directly distracting or even annoying him. In menus, sound gives an immediate feedback to user interaction when clicking a button or scrolling over a selection of options. In games and movies, sound is utilized to present a calm situation, to increase the excitement right before an epic battle and to accentuate the fighting itself. Horror movies are heavily based on music that increases excitement, as the many scenes are so dark that nothing in particular is noticeable. But the viewer expects that something could happen any moment because the beat of the background music accelerates. In games, sounds are generally used as an anchor for recurring events. Famous examples are the happy tune when a team won a multiplayer match and the frustrating sound when one's own character died.

Interactivity

Interactivity is the difference between watching an appealing movie with astonishing visual and acoustic effects and playing a game. In a game, the player has the possibility to interact with an avatar, with the environment or with other players. The player is able to decide for himself when and what he or she wants to do. The player's avatar and environment react accordingly. Despite open world games, which try to offer as much freedom as possible to the player, many games have a limited set of actions and restricted freedoms in which the player is able to operate. Nevertheless, even a small amount of interactivity could increase the fun in comparison to the passive role the user takes when passively watching a movie.

Multiplayer

Many activities are more enjoyable when experienced together with other people. In sports, hobbies or work, cooperating with or competing against other people may increase the motivation significantly. Playing games together with other people follows the same principle. Different gamification parameters as they may look when investigating the multiplayer, ego-perspective shooter *Word Domination* [146] are shown in Figure 3.5. As described earlier, *Word Domination* is a multiplayer online game where the players of two teams play against each other. They move through a given map and try to defend strategic positions while shooting with bubbles at other players. If someone gets hit or if someone takes over a strategic point, the players have to answer knowledge questions about a given topic. A team giving correct answers can act faster and get less time penalties. *Word Domination* can therefore be considered as a highly gamified quiz. In comparison to the initially digitized quizzes of Figure 3.2, it would get high scales in a number of categories. *Word Domination* is fully digitized, has a colorful 3D graphic and a high interactivity with its world and other players. But in contrast to a typical computer game, it has no sound effects and no achievements.

FIGURE 3.5: High amount of gamification in the *Word Domination* Game.

Achievements

Achievements are a virtual representation of a success. They are mostly used to motivate players to seek for further tasks besides the main goal of the game. Usually, achievements are *locked* and can be *unlocked* by reaching such a goal. These goals could include reaching a highscore or visiting several locations within the game world. Achievements typically indicate what the player has to do to unlock them. Achievements are a popular gamification element, they are used to improve the user experience with the software at relatively low costs [88, 103, 150]. They are also used by web communities to motivate their members to participate more frequently. E.g., Stack Overflow, a software development forum, uses achievement badges for users who are particularly helpful. These badges are then visible on the user's profile page. Wearing such a virtual badge can also increase one's professional reputation in the real world.

3.3 Basic Model

A few years ago when mobile devices, especially smartphones, began to spread among students, Audience Response Systems (ARS) had already existed for some years. At the University of Mannheim, a Java implementation for pocket PCs (WIL/MA) [126, 177] was already used in computer science for several years. This system was quite complex to set up, as it needed a dedicated wireless network, and the devices had to be distributed before each lecture. At this time, the aged hardware of those devices and the upcoming mobile technologies gave the impulse to rethink the original approach. Knowledge questions and paper tests were still a common element in everyday lectures, and the existing system's hardware could not satisfy the requirements anymore. Therefore, the task was

to improve the already used didactic method with a new solution by using the emerging technologies.

The task was to support the lecturer as well as possible. Only equal or less effort and time consumption than with analog methods were acceptable. According to Section 3.2, the proven knowledge checking methods first had to be digitized and accelerated. Performing a quiz should take place in three phases, as shown in Figure 3.6: A preparation phase where the lecturer creates the quiz, the actual execution of the quiz in the classroom and the discussion of the results. This approach was initially introduced, explained and improved in already released publications [4–6].

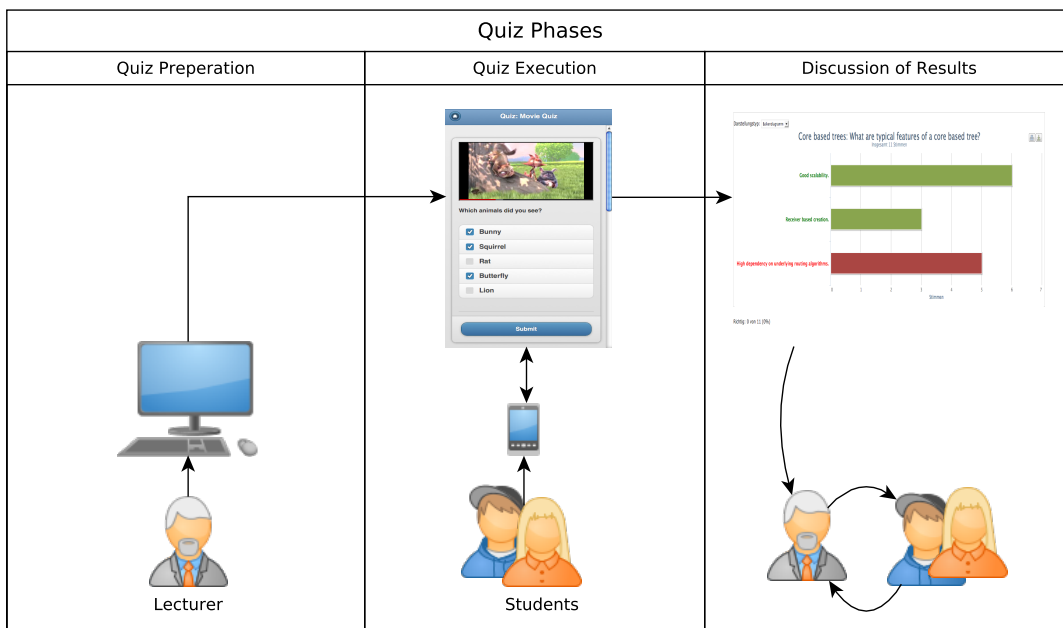


FIGURE 3.6: Basic process of the *MobileQuiz*.

3.3.1 General Conditions

When creating a new approach to support knowledge quizzes in large classrooms, it should be as simple as possible. Regarding the quiz application itself, the main goal was that students could use their own cell phones and other mobile devices to participate in the quizzes. One reason was that only a few students brought their laptops to the classroom, and it should be avoided to hand out additional devices to a large group of people. In case of a permanent under-representation, it would have been possible to supply inexpensive smartphones to the students that did not have any kind of suitable mobile device. Smartphones have the capability to access the Internet via a wireless LAN network or the mobile network of the phone provider. To support a wide range of mobile devices the quiz software was designed using web technologies instead of a native application (app). This should enable students to use their laptops and other web-enabled devices, too.

The resulting quiz system is focused on practical use, and therefore the approach had to consider further conditions. Beside students, the system had to be used by lecturers who created and maintained the actual content and initiated the quizzes during their

lectures. The final system had to be user-friendly. A small and clear data model with plain use concepts was a key condition for that. Lecturers had to be able to learn and use the system without external help. It therefore had to be self-explanatory.

Furthermore, the operation of the new application had to be ensured for a long time perspective. Unfortunately, the university is already full of applications written by former students and employees, which cannot be maintained any more. These applications are destined to die as soon as they cannot conform with an upgrade of an underlying system anymore. The new concept had to take this into account and had to try to integrate itself as much as possible into the already existing university infrastructure.

Besides the needed long-term resources, the development effort had to be considered, too. The project started with very limited resources, therefore it was not possible to develop an uncommon kind of software. Using widely spread programming languages, common libraries and design patterns would enable a fast development process and an acceptable base of programmers.

3.3.2 Concept

The basic idea of the new concept was *lightweight*. The database structure and the application on top of it should be as lightweight as possible. A web application, designed for smartphone browsers, would avoid the installation of new software by the students. Although a native application, once installed, can be opened faster and is usually more convenient to handle on a phone, it is far more effort to develop and to provide a separate version for every available type of phone and operating system. A web application, in contrast, is easy to develop and can be executed on many different phones and other devices. But native applications (apps) provide a set of benefits which are difficult to compensate. After a native application is properly installed and configured, it is easier to start in contrast to opening a web page. Starting an application is usually done by simply clicking on its symbol, and it offers a very good user interface for touch-sensitive devices. Typical web pages were not optimized to be used in this way, and therefore were very inconvenient to handle on a small phone. Furthermore, data for a personal authentication could be stored directly on the phone.

To compensate the lack of these advantages, our concept followed mainly two strategies: Adjusting the experience to native applications as much as possible and forgo the functionalities which could not be compensated. Code libraries already offered a surface layout equally to popular native applications. A fast access should be realized by a QR code, which encodes the address to the application's web page in a machine-readable way. QR codes were already commonly used in advertisement and most of the smartphones were equipped with respective QR code reader software. However, providing a secure and fast authentication did not seem to be realizable without a remarkable effort and therefore excluded. Lecturers, on the other side, will not administer the quizzes with their smartphones. They would rather use their laptops and perhaps the built-in PCs of the lecture halls. To prevent an additional installation on these computers, the idea was to increase the functionality of an already existing application. The learning management system ILIAS is widely used to support teaching and learning in many courses at our university. Every lecturer and student has an account by default, and the majority of them is very familiar with it. For that reason, it was decided that the quiz administrator functionality should be implemented as a regular plug-in to this system.

As a part of the productive e-learning system at the university, the long-term support would be done by the university's IT department. Additionally, the plug-in does not have to cover user management and authorization issues, as these would be covered by the main ILIAS application. As described in the requirements for the new system, the usability of the administration for the quizzes should be so simple that lecturers are able to use them without a profound introduction. They should be able to prepare a quiz for their next lecture by themselves. But furthermore, a quiz should be producible so fast that this could be done ad-hoc during a lecture. The user interface thus had to be simple and clean.

As the approach was designed to mainly improve knowledge quizzes, the data structure could be optimized to it. It is based on the concept of choice questions and simple input questions. The application would perform questions where the student could choose between a selection of possible answers, and input questions of limited size. The input questions were limited to numbers, as it was assumed that students could be annoyed by entering long text on the little touch screens of their devices. Numbers could be entered quite comfortably by shifting a slider to the appropriate value. The type of questions was therefore limited to answers that could still be handled very easily on a small mobile device.

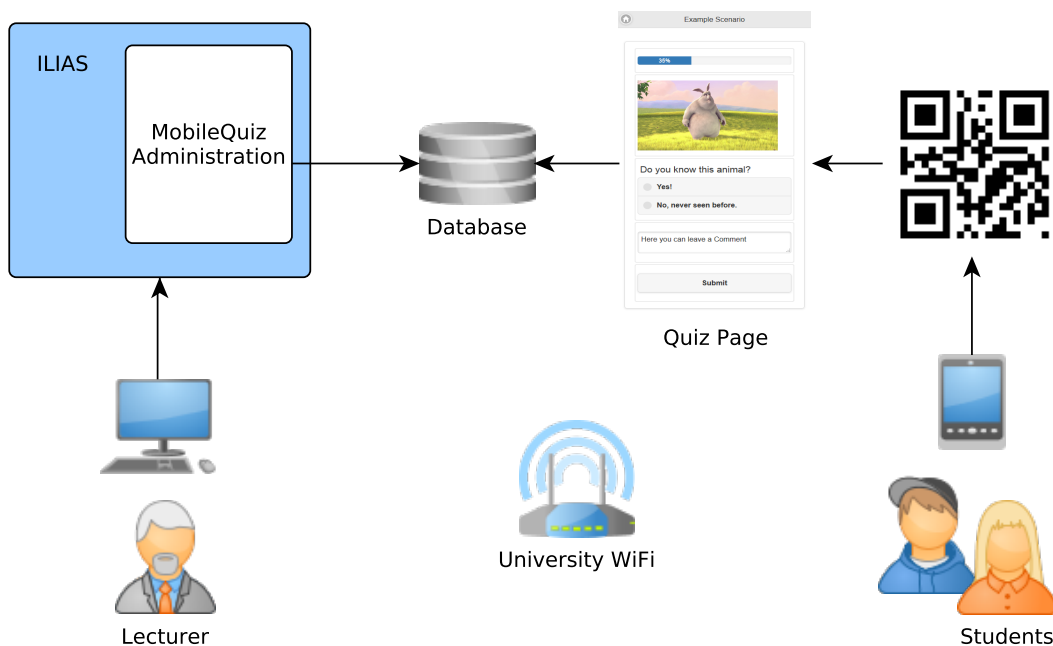


FIGURE 3.7: Basic concept of the *MobileQuiz*.

Figure 3.7 illustrates the basic concept. The lecturer creates a quiz as a learning object within his or her online course in the university's learning management system. In preparation of the next lecture, he or she uses a personal computer in an office. The questions and answers are stored in the system's database. When opening the quiz during the next lecture, the students use their mobile devices to scan the QR code and open a web page through the university's WiFi Internet. They don't need any kind of authentication and do not have to enter ILIAS, although the questions and choices are taken directly from the ILIAS database. The opened page then shows the quiz in

the familiar design of a native smartphone application. After the submission of all the answers the lecturer opens a result page to discuss the result.

3.3.3 Data Model

The data model contains five elements to store all relevant information. Figure 3.8 shows how they are related to each other. The *quiz* represents a concrete quiz with questions and answer choices. Several *questions* belong to one quiz. Each question may have several answer *choices* related to it. Every time a new instance of this quiz is used in a lecture, a new *round* is created. For every user and every choice, an *answer* is created, it stores the information which choices the participants have chosen.

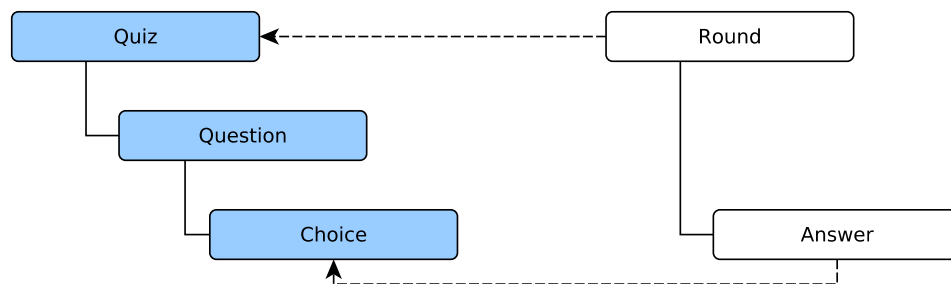


FIGURE 3.8: Simple data model of the *MobileQuiz*.

Quiz: The quiz holds meta information, like the quiz’s title and description. Moreover, it is the link to the ILIAS framework. The information about the quiz is also stored within ILIAS tables. ILIAS uses it to handle the *MobileQuiz* as a proper learning object.

Question: A question always belongs to one quiz. It only holds the information of the question’s text and type. The text is the displayed question. The type specifies how the question and choices should be rendered. The three types are: *single choice*, *multiple choice* and *numeric slider*.

Choice: A choice always belongs to a specific question, and a question can have multiple of choices. When the quiz is rendered on a phone, a choice is presented as one possible answer. Therefore, a choice stores information about the displayed text and a value of its correctness. This value can be *true*, *false* or *neutral*.

Round: A quiz can be executed many times. Every time a new quiz is created, a new round is started. Thus, several independent rounds of the same quiz can be performed independently. A round therefore stores the relation to the quiz it belongs to as well as a start and an end date.

Answer: Answers are used to store the information about the student's actual answers. They always belong to a certain round and therefore to a given quiz. For every round, choice and participant a new answer is created. It holds the information whether it was chosen or not. A numeric type question also stores the actual numeric value of the student's input. Answers are counted to create the result charts for every round.

3.3.4 Basic Execution Process

When executing a quiz, the actual technical process is very straightforward. As depicted in Figure 3.9, the QR code is created after starting a new quiz round. When it is displayed to the students, they scan it and read the encoded URL. The URL is then opened with a browser. A server-side user session is created, and the data to render the quiz is sent to the phone. Students click their favored answers and submit. The submitted information is sent back to the server and stored. The information is summarized and can be displayed in the form of colored result charts. Afterwards, the quiz can be closed.

This process can be executed faster or slower. The quiz could be prepared ad-hoc within the lecture for only one question and be closed five minutes afterwards. Or, it could be active for several weeks, to be done at home. The QR code could be printed on an exercise sheet, and students could test their knowledge while they resolve their weekly tasks. But nonetheless, the communication is always in one direction. After submitting a quiz, there is no further communication between the student's device and the server.

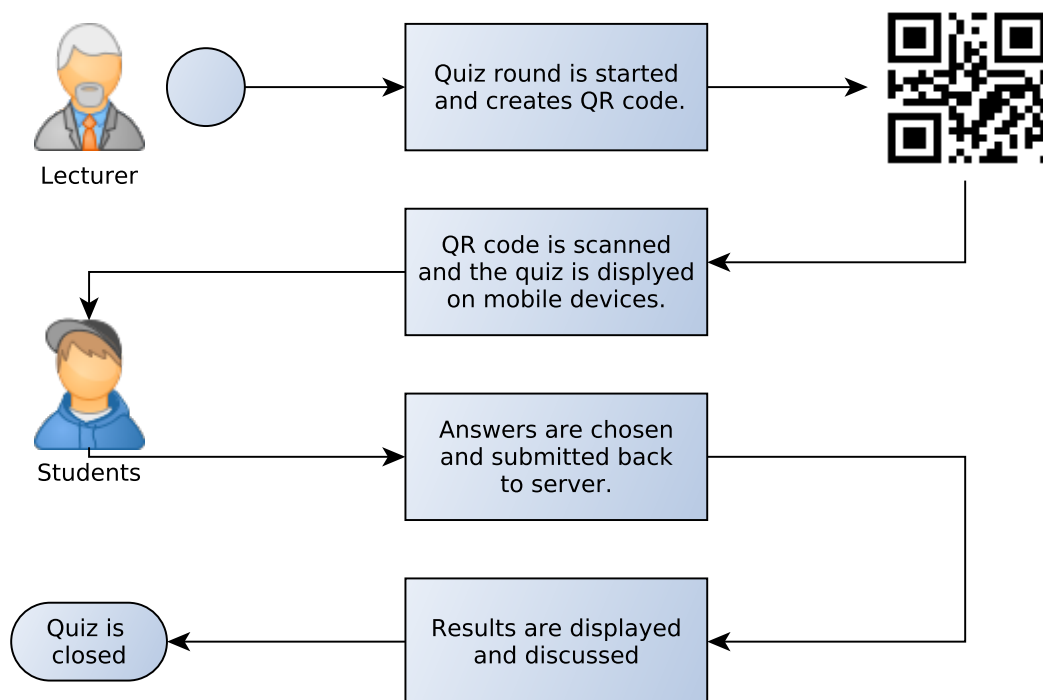


FIGURE 3.9: Basic process when using the *MobileQuiz*.

3.4 Advanced Model

Unfortunately, the basic model is not able to describe the full extent of the lecturers' requirements. But instead of extending the simple model with a number of smaller improvements, a fundamentally new model was designed from scratch. This new model takes the diversity of teaching scenarios into account from the beginning. This means an expanded view on the teaching scenario's life cycle, as well as a new structure for the data model. The concept was initially presented in Schön et al. [9] and explained in further detail in Schön et al. [11] and Schön et al. [14].

After expanding and abstracting the view on the concept of the *MobileQuiz* in particular and any ARS in common, the new model showed similarities to the concept of a classical board game. A game usually consists of a card board and different gaming pieces. These pieces are mostly tokens in several colors, cards with text, and resource coins in different values. Figure 3.10 shows examples of several kinds and shapes of pieces on a colored board. In addition, a board game has a set of rules, which describe the way the pieces have to be handled.



FIGURE 3.10: “The Settlers of Catan”TM game with board and various game objects.

When taking an abstract view on the teaching scenarios of ARSs, these elements can be identified, too. The screen of the students' mobile devices can be seen as the underlying board. Students interact on that surface with the different objects. Like the different gaming pieces in the board game, the objects in the teaching scenario are the questions, answers and buttons on the screen. As in the board game they differ in size, color, shape and further attributes. Some of these objects are click-able, whereas others can only be read. Further objects allow textual input, whereas others only allow numerical input.

Besides the objects and the board itself, a game has *rules*. Rules describe the logic behind the game and the way players interact with the game elements and with each other. This is also valid for ARS teaching scenarios. Usually, these rules are part of the background logic of an implemented ARS. A rule of a quiz may be that the questions should disappear when the student submits the answers.

Therefore, in an abstract perspective, board games and ARS teaching scenarios can both be described with the same abstract schema. The new model should be able to describe scenarios beyond pure quizzes. Beside the functionality of the basic model, it should enable complex scenarios up to live experiments and marketplace simulations. As a result, it should create didactic benefits, beyond the usage of plain quizzes.

3.4.1 General Conditions

The conditions of the advanced model are significantly different from the basic one in Section 3.3. Since the first model was focused around a very lightweight approach, the successor model moved this requirement into the background and mainly focused on general model validity and the possibilities for customization.

Nonetheless, some conditions remained similar. The new approach is also based on the students' mobile devices and an existing WiFi Internet network. The basic approach took benefit from the new and increasing distribution of mobile devices among students. With today's technology level, the new approach could assume an availability of mobile devices of almost 100 percent. But the variety of different devices is similar to before. For this reason, the advanced approach used a similar mobile web application strategy as the basic one.

The basic model focused on an application, which has a high practical usability, so that it can be utilized by lecturers without external support. The advanced model still should be easy to use, but a more general validity of the model and its customization has a higher priority. With the disadvantage of a more complex data structure, the new data model should not only describe quizzes and audience feedback, but also depict further teaching scenarios. These could reach from message boards with a very reduced kind of user interfaces up to very complex marketplace simulations. The data model should have an universal validity for almost any possible teaching scenario. This gives the lecturers more freedom to customize their individual teaching scenarios exactly to their needs. But because of the raise in complexity, external support has to be considered. Especially in designing and modifying scenarios, lecturers should be supported by a technical and didactic specialist.

The architectural assumptions were not focused on long-term viability as the first model. Therefore, the integration into the existing university's infrastructure and software landscape were not intended. Despite the advantage of easy access for lecturers, the integration showed some major disadvantages, too. The implementation into the infrastructure of Mannheim prevented the deployment at universities with different settings. Universities not running an ILIAS system were not able to use the earlier *MobileQuiz*. In fact, lecturers and research colleagues from outside the University of Mannheim could not get access to the local system. Due to the increasing collaboration with international colleagues, the new implementation should take this into account. Therefore, it should be a stand-alone tool with interfaces to optionally connect it to existing university infrastructures. With that, it should be as independent of third-party software as possible.

The available resources and timeline for the new approach were higher than for the first application. The prototype, which based on the new advanced model, could therefore use more complex technologies. The applied programming language, software framework and

libraries could be novel to the project team the time framework allowed more training time. In addition, didactic experts were incorporated in the project from the beginning to ensure that the new tool would be useful for didactic research.

3.4.2 Concept

Whereas the main motto of the basic model was *lightweight*, the main motto of the advanced model became general usability with *customization*. *Customization* is the last step in Section 3.1 after digitizing, acceleration, gamification and generalization. Therefore, digitization and generalization have to be done already before the new model can be successfully implemented and tested.

Despite the different priorities, several aspects of the basic model's concept remained the same. Among these are the usage of the students' mobile devices, the focus on university lectures and the implementation as a web application.

As already mentioned, the new and more generalized concept is based on the idea of a classic board game. As mentioned, a board game is considered as containing a board, a set of gaming pieces and rules, which describe the handling of the elements. Abstracting the gaming pieces then can be seen as gaming objects with different sets of attributes. Figure 3.11 illustrates how these elements are related to each other in that abstract view. *Objects* and *rules* are bound to a specific teaching *scenario*. *Attributes* belong to an *object*, and every *rule* consists of a set of *conditions* and *actions*.

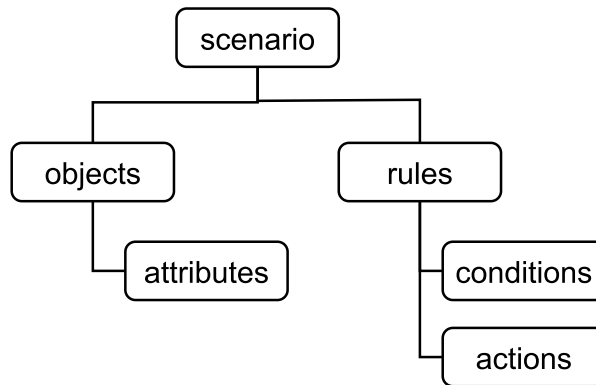


FIGURE 3.11: Elements of the advanced model.

Objects are elements related to each other and holding information in the form of *attributes*. Compared to the classic board game in Figure 3.10, the wooden piece in the shape of two towers symbolizes a city and gives its owner a specific resource income and victory points. It can be seen as such an object.

Every object holds a set of *attributes*, which store the actual data. The wooden piece example possesses the attributes' shape of the piece, its color, the amount of victory points and its costs. In a teaching scenario, an attribute can be the text of a question, a check box for an answer or a button for a submit. Every information about an object is stored within one of its attributes.

Whereas attributes are only connected to their object, objects can also be connected to each other. In the board game, a city can consist of a church piece, a city-wall piece and a piece representing the towers of a cathedral. A city therefore consists of three objects altogether. Compared to the quiz example, a question can have a set of possible answers objects, each with its own attributes. A further benefit of this description with objects and attributes is that the model can be coded with a object-oriented programming language more easily.

A board game usually is supplied with a small rule book describing how the game is meant to be played. It contains information about how the game board and pieces have to be arranged, how the player should interact with them, and how he or she should interact with the other players. In the abstract view of the advanced model, rules should adopt this very same task. They should describe the time, order and conditions in which the objects and attributes should be interacted with. Every rule consists of a set of *conditions* and *actions*. Conditions describe the circumstances under which the rule is activated, while actions explain the actual task that will be performed. This idea covers very simple rules with only one condition up to very complex rules, when several conditions have to be met before the complete user interface changes its layout. In a simple example, a button-click could trigger a player's counter to be increased by one. In a more complex one, a sum of given attributes of all players has to exceed a given value. In another example a player with less than five correct answers could get a warning message on the device's display.

Every object and every rule belongs to a defined *context*. The context of *objects* defines the number of objects which are necessary in a given scenario. An object may contain the context *game* or *player*. Objects within a game context only exist once in every game. They are used to store information, which is valid for all players, e.g., a time counter or the sum of all players' inputs. Objects with a player context are created separately for every player. These can be the specific answers of a test question.

Rules need more contexts than objects. In addition to *game* and *player*, they have the contexts *object*, *myObject* and *myPlayer*. The contexts of rules determine how many attributes are taken into account when checking the conditions of the rule and when performing the actions.

3.4.3 Data Model

Rules with their conditions and actions are stored in the database once, they will not be changed throughout the phases. But the objects and attributes have to be stored separately for every main phase (scenario, quiz and game). Figure 3.12 shows the data structure of all the elements throughout these three phases. The blueprint of the scenario is implemented with *objects* and *attributes*. They can be understood as objects of an object-oriented programming language as they specify the structure of the scenario. In the quiz phase (phase two), specific entities of these objects are created (with *entityObjects* and *entityAttributes*) which contain specific information about this quiz. For each of them, at least one instance of a *gameObject* and *gameAttribute* is created at the start of the game phase. Additionally, when a *gameObject* can be assigned to a *player*, it has a reference to this specific player.

Objects and *attributes* store information about the general structure of the scenario (e.g., if it is a multiple choice test or a survey), *entityObjects* and *entityAttributes* store

information about a concrete topic (e.g., questions about elephants), and *gameObjects* and *gameAttributes* hold the data of one game and player (e.g., a player's answers to this test).

The information for the *result* phase and *analysis* phase are also stored in the *gameObjects* and *gameAttributes*.

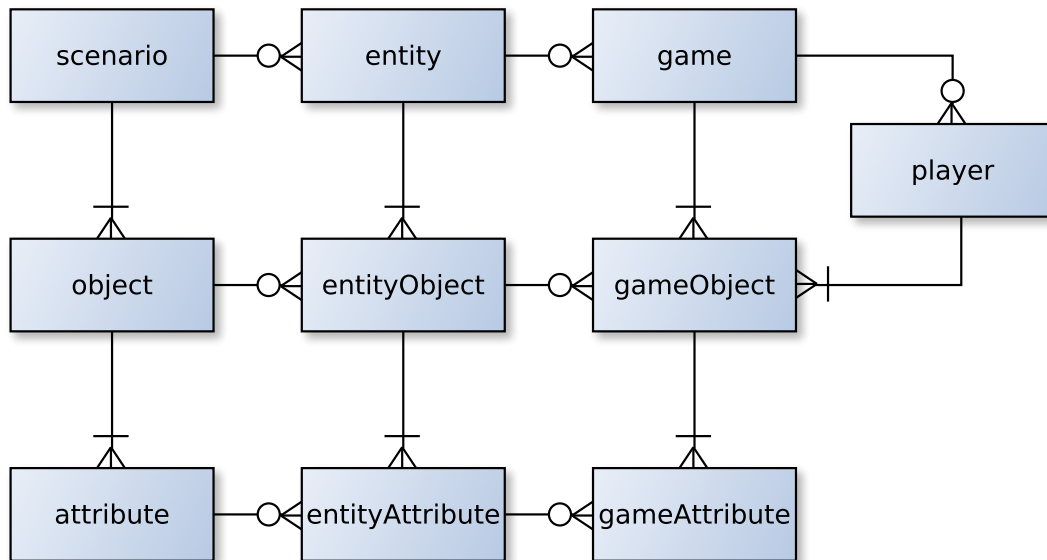


FIGURE 3.12: Core data model of the advanced model.

Scenario: The *scenario* element is the basic anchor of every teaching scenario and stores its name, status and description. The name should be a unique identifier to easily understand the scenario's purpose. The status shows if the scenario is active and can be used.

Object: Every *object* belongs to only one scenario, whereas a scenario can contain different *objects*. Objects store the information about the shape, structure and visibility of the scenario. But the data itself is stored within the attributes. Like the scenario, objects also have a name and a description. They store the context information and state if the object belongs to a player or to the entire game. A *chart* flag states if the object should be considered when creating the result charts at the end, and how they should be visualized. Possible chart types are bar charts, pie charts and plain text listings. The *mode* and *editmode* state the visibility of the object for the student during the game phase and the lecturer during the quiz phase. Valid values are *read*, *write* and *hidden*. Hidden objects may not be visible, whereas readable objects are visible and may be edited, but may not be created again. E.g., an object with a general information text may be modified, but cannot be created again as the scenario only allows one of them. In contrast, writable objects may be created more than once. E.g., questions and answers may be writable as the quiz may contain many questions with different sets of answers. Every object can be related to another object, like answers are related to a specific question. In particular, every object can store a reference to its father object. The model currently only supports hierarchical relationships. Every object may have many different son objects, but only one father object.

Attribute: *Attributes* are the blueprint for the elements, which store the actual data. They have a name and a description to simplify the handling for the lecturer. Like the *objects*, they also have a *mode* and *editmode*. Within objects, *mode* and *editmode* state if this object could be created separately. But within attributes, *mode* and *editmode* only state if the information can be seen and modified. Attributes can never be created separately, as they always belong to an object. Besides the actual *value*, an attribute may also have a *default-value*. Like the object, an attribute may have a *chart* flag, which shows if the value of this kind of attribute should be considered as title or data for a chart in the result phase. Finally, every attribute has a *type* which states the kind of information it represents. Depending on the type, the attribute's appearance on the students device differs. Possible types are button, text, char, num, slider, bool or image. Table 3.1 describes the most common types in more detail.

TABLE 3.1: Common types of attributes with their final visualization.

type	content	visualization
button	string	A button with the text value written on it.
text	string	Depending on the mode, it is displayed as a label or text input field. E. g., for comments.
char	string	Depending on the mode, it is displayed as a label or single line input field. E. g., for single word inputs.
num	integer	Depending on the mode, it is displayed as a label or single line input field which only allows numeric values.
slider	integer	An HTML slider for numeric values.
bool	boolean	An HTML checkbox.
image	picture	If writeable, an upload functionality for pictures is rendered. If readable, the picture is displayed.

Entity: An *Entity* contains the data about the actual topic and represents a certain quiz. An *Entity* is a concrete representation of a blueprint scenario. Therefore, it contains a quiz title and a short description. Whereas existing blueprints can be used by everyone, *Entities* are owned by lecturers and can only be edited by them.

EntityObject: The *EntityObject* is the *Objects* representation in the quiz phase. It contains no important information on its own and mostly is a container for the *Entity-Attributes* in the quiz phase. The only unique information it contains is a number for the sort order on the board.

EntityAttribute: *EntityAttributes* contain the topic-related data of the actual quiz. If the quiz is a multiple choice test, the actual question and answer texts are stored within these attributes. Therefore, *EntityAttributes* include the values given by the scenario's blueprint, and entered by the lecturer for a specific quiz. Remaining information about the *EntityAttribute* is available via a reference to an *EntityObject* and the *Attribute*.

Game: A *Game* is the execution of a specific quiz round within the classroom. The *Game* has the data about a start and end date of this execution. The status of a game defines if the *Game* is active and can be accessed by the students, or if it can be used

repeatedly. Every game has a unique URL by which it can be accessed by a browser. In addition to this URL, a shortened URL is stored in the *Game* element which is also used for the QR code generation.

GameObject: A *GameObject* is the concrete representation of an *EntityObject* in an actual quiz round.. Similar to the *Object*, it stores only little relevant information. It contains a reference to its father-*GameObject* and a reference to a *Player* (the student it belongs to). Only a handful of *GameObjects* are related directly to the game itself as they are used for overall objectives. Most of them belong to a participating student, represented by a *Player* element. *GameObjects* also contain a separate mode. It is utilized to individually render the board on the students' devices. E.g., after the submission of answers in a quiz the objects should not be edited anymore, and the submit button should be hidden for this student.

GameAttribute: The *GameAttributes* are the container for the information generated within the classroom. They store the values entered by the students and are therefore used as source for the results later on. Possessing their own mode, they can individually be rendered like the *GameObjects*. In addition, every *GameAttribute* has a timestamp of its last modification for statistical analyses.

Player: Every student who participates in a quiz round is represented as a *Player*. Besides the set of related *GameObjects*, every *Player* has a “banned”-flag. This can be used in extended teaching scenarios with a potential for misuse. Thereby, the risk to be disturbed by malicious students can be decreased. In addition, the last activity of every *Player* is stored to enable an overview about its progress. If a significant amount of participants still show an ongoing activity, they might need more time to finish this scenario.

3.4.4 Rules

Rules describe the way how the elements of Section 3.4.3 should be handled in the execution process during the game phase. They are defined at the beginning, during the blueprint phase, and are not modified or copied in later phases. Every *Rule* contains a set of *Conditions* and *Actions*, as seen in Figure 3.11. Beside these, every rule has a given context in which it should be executed. To guarantee transparency, *Rules* also have a unique name.

Context of Rules

The *Context* of a rule determines the set of *Attributes* it works with. The *Context* varies from one specific input, up to all attributes of the current game round. Rules are executed for every game round separately and cannot affect each other. The conditions and actions always refer to a determined *Attribute*. When the rule is executed, the *GameAttributes* of that *Attribute* are gathered and used. E.g., a rule with the context *game* gathers all instances of an attribute of the game at once and checks the defined conditions against them. If the condition is satisfied, the rule becomes active. In another

example, the sum of all numeric answers is higher than a given threshold. In contrast, a rule with context *player* only gathers the attributes of this very *Player* and checks the conditions and performs the actions.

Originally, only the contexts *Input*, *Object*, *Player* and *Game* were implemented, as it seemed possible to describe the logic of every theoretical scenario by using them. Thereby, a *Player* context rule was executed for every *Player* himself, every time it was activated. Similarly, an *Object* context rule was executed for every *Object* of the game, every time it was activated. Unfortunately, this caused a significantly high amount of computing power and stressed the servers in scenarios with a moderate amount of complex rules and more than fifty participants. In consequence, the contexts *MyObject* and *myPlayer* were introduced to limit the active rule on the *Object* or *Player* who triggered it. Figure 3.13 shows which attributes are considered in the different rule contexts, when a student activity is performed.

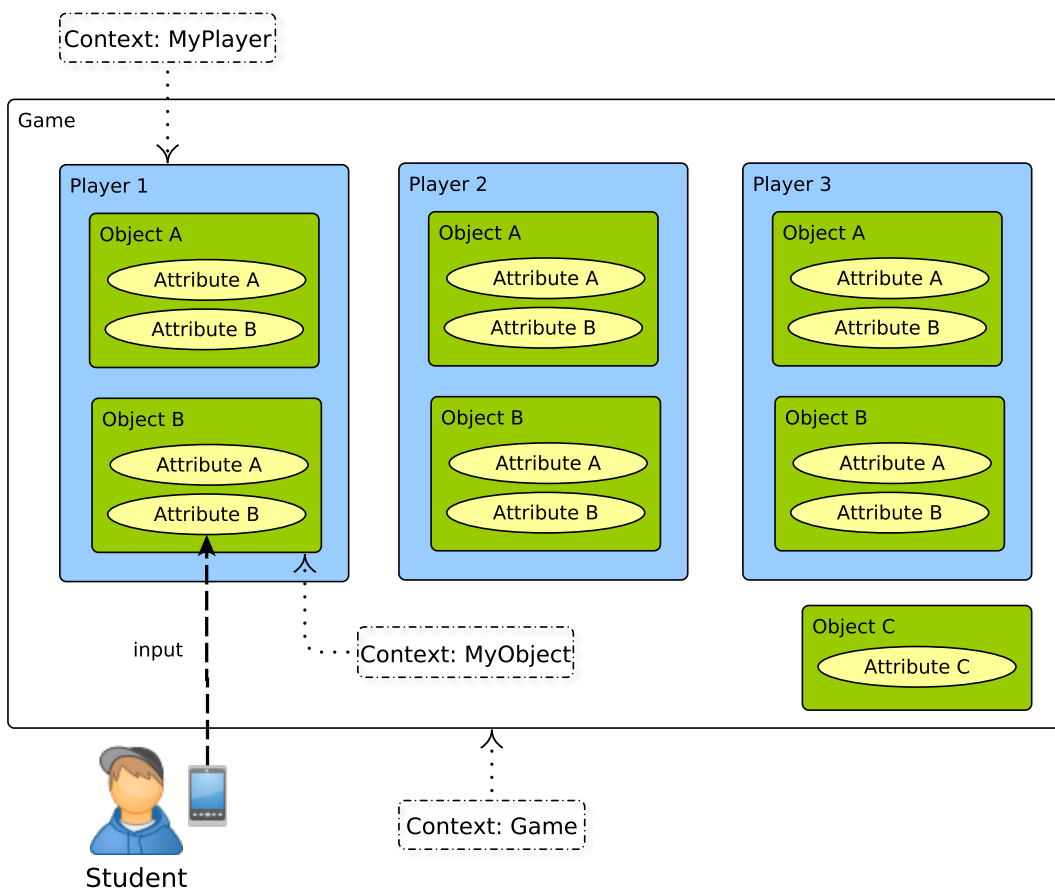


FIGURE 3.13: Different contexts of rules in the advanced model.

Although the set of *GameAttributes*, that should be considered, is initially determined by the context, the rule is not fully limited to it. Figure 3.14 depicts the process how the rule gathers the necessary *GameAttributes*. Conditions look only at the given context. If no suitable *GameAttribute* can be found within the given context, the rule is not activated and quits. In contrast, if an action does not find any attribute in the given context, the context is extended to the next level until suitable attributes can be found. If an object context rule does not find any suitable *GameAttribute* within the given

GameObject, it extends the context to *Player* and looks for suitable *GameAttributes* in all *GameObjects* belonging to this *Player*. If no *GameAttributes* can be gathered either, the context is extended to all *GameAttributes* of this *Game*. Therewith, it is relatively easy to describe complex behaviors efficiently with only little information.

Input: A rule with context *Input* only considers the input of the actual activity of the student. This is only the one *GameAttribute* which was triggered by the student. In many cases this will be a button or the value of an input field. The rule then checks this very value for the given condition. E. g., a popular condition is to check if a button was clicked. As seen in Figure 3.13, the activity of the student always affects only one *GameAttribute*.

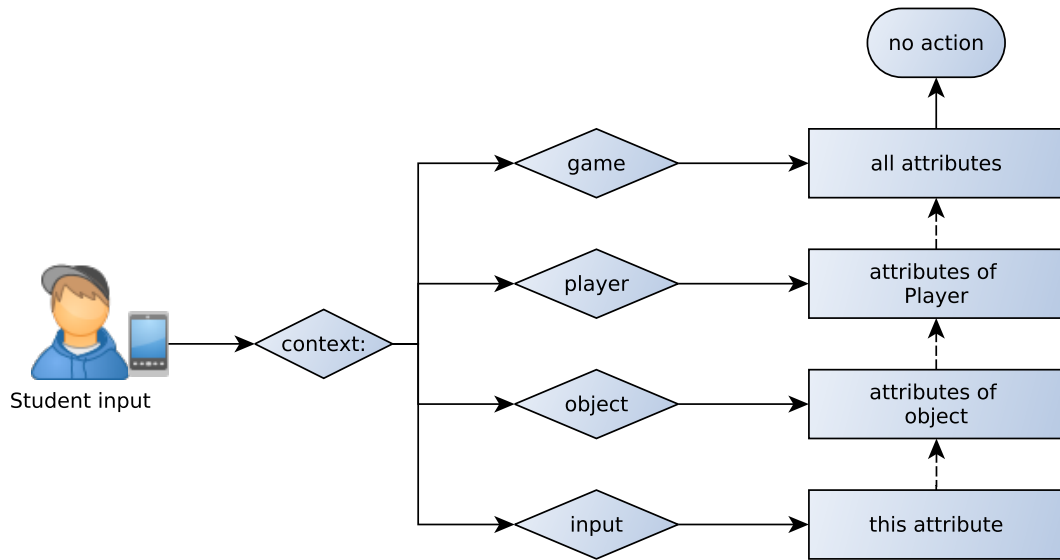


FIGURE 3.14: Processing of rules of different contexts in the advanced model.

Object: Context *Object* is used to perform rules in the scope of individual objects. The rules in context *Object* are checked and executed for every *GameObject*. When dealing with every *GameObject* of the whole *Game*, performing this rule can be very time consuming. It can be used to check if the input to a given question is correct and to highlight questions with wrong answers. In fig. 3.13, an *Object* context rule would deal with all six objects successively as all contain an Attribute B.

Player: Similar to the context *Object*, a *Player* context rule deals with the *GameAttributes* of every *Player*. This can be used to hide all questions of this student, when he or she submits the answers. Like in the *Object* context, the execution of this rule may be very time consuming in scenarios with many participants. In Figure 3.13, a *Player* context rule would deal with the *GameAttributes* of each *Player* successively.

Game: The execution of the rules in the context *Game* is very straightforward. The rule directly gathers all suitable *GameAttributes* of the whole *Game*. Thereby it makes no difference, if these belong only to the *Game* or to *GameObjects* belonging to *Players*.

In Figure 3.13, if a *Game* context rule is related to Attribute A or Attribute B, it would deal with all of them at once. A rule related to Attribute C will also deal with all Attribute C, but because this only belongs to the *Game* itself, only one *GameAttribute* is affected.

MyObject: The execution of the rules may be very time consuming, when every single *GameObject* has to be executed for itself. But in many possible scenarios it is sufficient to deal with the one *GameObject*, which triggered the rule. The rules of the *MyObject* context deal with a much smaller number of *GameAttributes* which is significantly faster when performing a game round. In Figure 3.13, this rule would therefore only affect the *gameAttributes* of Object B of Player 1.

MyPlayer: Similar to the *MyObject* context, the *MyPlayer* context rules only deal with the *GameAttributes* of the current student. In many cases, the consequences of a triggered input only affects the triggering student himself, so that the other *Players* may not be affected. In Figure 3.13, this rule would therefore only affect the *gameAttributes* of Player 1.

Conditions

Conditions specify the status in which a rule is activated. They are structured in three parameters: an *attribute* which should be checked, an *operator* which describes the method to check and a *value* against which the *attribute* should be checked. The value may be a plain value or a reference to another *Attribute*. In the latter case, the value of one *GameAttribute* is checked against the value of another. Table 3.2 shows a selection of possible conditions.

TABLE 3.2: Set of conditions used in rules of the advanced model.

Symbol	Condition	Description
==	equal	Checks if the value is equal to a reference value.
!=	not equal	Checks if the value is not equal to a reference value.
lt	less than	Checks if the value of the value is smaller than a reference value.
gt	greater than	Checks if the value of the value is greater than a reference value.
involved	involved	Checks if the attribute is involved in the current activity.
empty	empty	Checks if the value of the attribute is empty.
notempty	notempty	Checks if the value of the attribute is not empty.
~~	similar	Checks if a value is similar to a reference value. E. g., 001 is similar to 1 and 'car' is similar to ' Car '.

Actions

Actions specify the operations that are executed after a rule is activated. Actions are structured similar to conditions. They also have an *attribute*, which can be modified, an

operator, that describes the method and a *value* which represents the source information for the modification. The *value* may also be a string or a reference to another attribute. For some operations this reference can be used on a large number of source attributes. As seen in Table 3.3, operators like *min* or *sum* use a set of source data to calculate the target value. Other operators are used to modify the layout on the board, they hide attributes or set them to writable.

TABLE 3.3: Set of actions used in rules of the advanced model.

Symbol	Action	Description
=	set	Sets the value to the value of the reference.
+	add	Adds the reference value to the attributes value.
*	multiply	Multiplies the value with the reference.
min	minimum	Search for the minimum value of all attributes of the given reference.
sum	summarize	Summarizes all values of the attributes of the given reference.
counttrue	count true values	Counts all values which are <i>true</i> in the attributes of the given reference.
countfalse	count false values	Counts all values which are <i>false</i> in the attributes of the given reference.
sethidden	hide attribute	Hides the attribute.
setwritable	enable writing to attribute	Enables the attribute, so that students can interact with it.

3.4.5 Advanced Execution Process

Unlike the basic execution process of Section 3.3.4, the advanced execution process is more atomic and repetitive. In the basic process, all student information (the answers) were sent to the server at once, when the student hit the submit button. In the advanced model, every information is send to the server separately, immediately after being entered. Clicking a submit button triggers the same kind of process as entering a text into an input field. Even if no information is entered for five seconds, an update request without a concrete modification is triggered automatically. As a result, the student's interface can react on events triggered by someone else.

Figure 3.15 shows how this process is executed. It starts either with a student's manual input or the five seconds auto refresh. The server-side receives the request and – in case of a data input – immediately writes it into the database. Then all *Input* context rules are performed successively, whereby the conditions are checked against the given input. After that, the rules of the remaining contexts are performed against the information of the current database status. If one of the rules is activated, it directly changes the data in the database. Afterwards, all necessary information is gathered from the database and sent to the student's device. The browser compares the received data with the visible data and updates the information that differs.

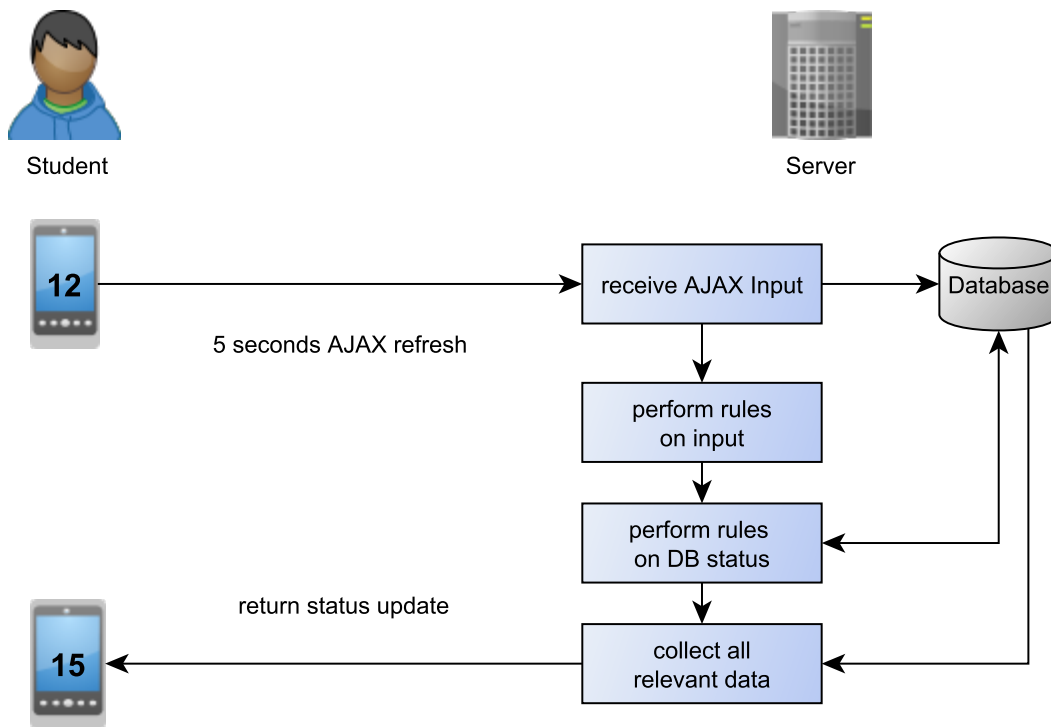


FIGURE 3.15: Advanced execution process showing the steps when new information is entered on the student's device.

3.5 The Five Phases of a Teaching Scenario

The life cycle of a teaching scenario can be divided into five disjoint phases. As listed in Table 3.4, these phases are the scenario *blueprint* definition, creation of an actual *quiz*, performing an *interaction* within a classroom, discussing the *results* of it and *analyzing* the efficiency of the overall scenario.

TABLE 3.4: The five phases with involved roles and locations.

#	Name	Description	Duration	Roles	Location
1	Blueprint	Blueprint definition of a teaching scenario	1 hour to 1 day	lecturer, specialist	office, meeting room
2	Quiz	Creation of an actual quiz of a scenario	10-60 minutes	lecturer	office, classroom
3	Interaction	Performance of a classroom activity	5 minutes to 1 semester	lecturer, students	classroom, at home
4	Result	Presentation of the result	5-15 minutes	lecturer, students	office, classroom
5	Analysis	Analysis of students behavior and scenario success	1 day	specialist	office

The experiences with the basic model of Section 3.3 showed that it did not cover enough phases of a teaching scenario life cycle as the implemented software had to be modified

repeatedly. Lecturers continuously brought new requirements and wishes for modifications to the developers' attention. It became clear that the model lacked general validity. The basic model only covered the phases two to four of Table 3.4. The process of designing a new teaching scenario and analyzing the overall quality afterwards were not considered. The advanced model of Section 3.4 took this into consideration, and the *Blueprint* and *Analysis* phases were added.

Figure 3.16 shows how the data model of the advanced model fits into these five phases. The blue boxes are similar to the basic model, the gray boxes are only available in the advanced one. *Scenario*, *object* and *attribute* define the blueprint. *Entity* corresponds to the quiz of the basic model and *entityObject* and *entityAttribute* contain the actual quiz data. A *game* is equal to a round, and *gameObject* and *gameAttribute* therefore contain the students input. All game elements are used for the interaction and result phase. In the final analysis phase, all elements of the other phases are used again.

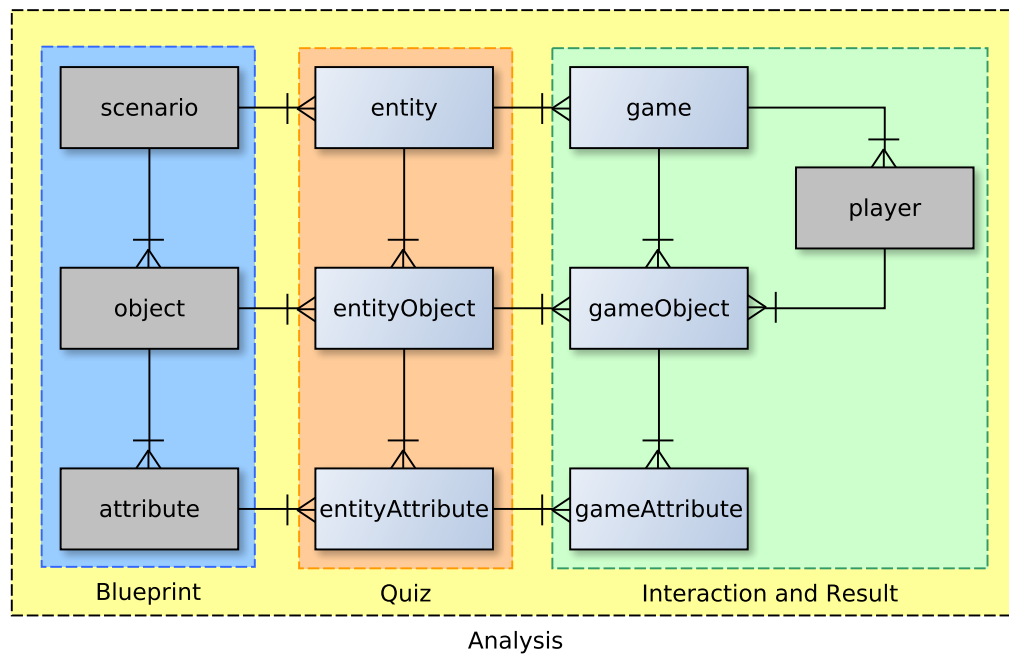


FIGURE 3.16: Data model of the *MobileQuiz2* in relation to the five phases.

3.5.1 First Phase - Blueprint

In the first phase, the initial blueprint of the teaching scenario is created. Before the scenario can be shaped, the learning goals of the course unit have to be stated, and the new learning scenario should be fitted into the overall concept of the course. According to the learning goals, the new scenario could be a personal knowledge feedback, an in-class message board, a live experiment in game theory or a feedback of the current talking speed of the lecturer. The lecturer has to define the elements and appearance on the students' phones, the interactions between the participants and the compilation of the charts he or she wants to present and/or discuss in the classroom. At this point, the decision is made whether the learning scenario consists of a single submit button or

a more complex set of buttons, check boxes, input fields and rules. When the scenario should simulate a complex marketplace situation in an economics class, it can be a high effort to describe this scenario on paper to another person. Therefore, this phase needs a fair amount of structural and didactic input. Scenarios that are very useful in a specific classroom setting could be devastating in a slightly different environment. Lecturers do not always have the experience in using such teaching scenarios and can easily design suboptimal elements. But mistakes in the scenario's blueprint are very difficult to compensate in later phases. Therefore, every scenario should be structured carefully in order to support the learning process of the students. It is recommended to create scenario blueprints together with a didactic expert. This expert could collect the experiences of foregoing scenarios and use this knowledge to support the lecturer in designing a well-fitted and didactically useful scenario.

If the scenario, for example, contains many buttons, figures and input fields because the lecturer wanted to cover a lot of things within one scenario, students could be overstrained by the full setting and unable to solve the task. In a small scenario of about 15 students, a message board may be an acceptable solution to get an overview of the learners' understanding of the course material. In contrast it may become impossible in large lectures with more than a hundred people. The number of results would simply not allow a fast and useful overview.

Defining a new scenario does take some time. It needs a productive environment, where lecturer and specialist are able to discuss possible variations of the scenario. The location for this will mostly be an office or a meeting room. The used computer therefore probably will be a laptop or PC and may be connected to a projector. The application's interface should be optimized for smaller laptop screens and low resolution projectors. As specialist and lecturer are able to work concentrated and patiently during this phase, additional explaining features would be available.

3.5.2 Second Phase - Quiz

After the scenarios blueprint is defined, it can be used to create a concrete entity of a quiz. This entity is called *quiz*, according to the basic model in Section 3.3. But as the new model supports a higher amount of customization, these *quizzes* may not be limited to classic knowledge tests. The lecturer takes the scenario template and fills it with actual content of a specific topic. If the teaching scenario is a knowledge quiz according to the basic model, he gives it a title and enters text for the multiple choice questions. He or she can only use the objects defined in the blueprint. Objects marked as *hidden* will not be visible. Objects marked as *readable* can only be edited, but further objects of this kind can not be added. E.g., an object representing the sum of the students' input may not be changed by the lecturer, as this may corrupt the scenario's logic. The individual attributes behave similar to the objects. Attributes marked as *hidden* are not visible to the lecturer but attributes marked as *readable* are visible, but can not be edited.

The second phase is usually done by the lecturer alone during the preparation of the next lecture. Usually the lecturer sits in a work environment like an office. The application's user interface of the quiz editor can therefore be optimized to a desktop PC with a high resolution monitor and a calm environment. The lecturer would have the time and possibility to read explaining texts and tool tips. Depending on the experience of the

lecturer and the complexity of the scenario, creating a new quiz entity can take 10 to 60 minutes.

3.5.3 Third Phase - Interaction

The concrete quiz entities, which now have an actual course-related content, can then be used to interact with the students. Technically, the objects of this phase use the prefix *game*, as according to the board game reference, a classroom game is performed. These interactions are similar to the game rounds of the basic model. During the course, the lecturer opens the application and presents the QR code of the quiz to the students. They scan it, open the URL in their browsers and join the quiz. The quiz now lasts until the lecturer closes it. Depending on the scenario, the students then submit an answer to one quiz question, or interact with each other through several tasks. Usually, the actual quiz interaction will be performed in a lecture room. These can reach from 15 people seminar rooms up to large lecture halls with several hundreds of seats. Therefore, the application should offer a user interface, which enables the lecturer to get a good overview of the current status in his or her room, and it should allow an easy access to all important features at once. The lecturer will be in a stressful situation in front of the classroom and will not read long explanation texts. In addition to the present students, remotely attending students should also be able to attend the quiz over a video stream. There should be no difference in students accessing the quiz from the classroom or a remote location. Besides this live in-class situation, a quiz can also be started at home, and students may access it alongside lecture preparation or post-processing. Every quiz entity is persistent and can be used simultaneously for many games in different lecture groups and at different times. Interactions can take a short period of time, e.g., five minutes for a small self-assessment test at the beginning of a course, or a long period of a whole semester for student exam preparations, surveys or course evaluations.

3.5.4 Fourth Phase - Result

In the fourth phase, the students' input can be displayed as *result* charts. Depending on the type of input data, the results are displayed as text, colored bar charts or summarized stacked charts. The type of chart and color can be defined during the blueprint phase. If no separate definition was provided, default colors will be displayed. The colors red, green and blue are automatically used for wrong, correct and neutral answers. The lecturer should be able to display the results of the current scenario with one click. Depending on the scenario, the students may not see the results of their classmates to stay uninfluenced. The results usually will be discussed by the lecturer and the students in the classroom directly after the quiz interaction, as the answers are yet present in everybody's mind. But depending on the scenario, they may also be discussed way after the time when the questions were answered. The charts thus have to be self-explanatory and should display enough information in a way that everybody is able to reconstruct the questions. If the lecturer has to jump back to the quiz again, this may cause confusion and displease him or her and the students.

3.5.5 Fifth Phase - Analysis

Finally, the fifth phase is the analysis of the scenario's validity, the students' learning behavior and further didactic research issues. Therefore, everything entered by the students has to be logged in separate tables. Besides the actual input of the students, which is used for the result charts, timelines of every information should be stored together with complementary meta data. This data could then be used to perform learning analytic research. To avoid privacy issues, no user-specific data should be collected, the data of the participating students should be collected anonymously. The analysis will mostly happen in an office environment by at least one specialist who is able to compare different data sets and draw conclusions. This could happen directly after a new scenario was tested for the first time, or at the end of the semester to sum up the weekly experiences. With these conclusions in hand one can go back to the first phase by modifying the blueprint according to the results, for further improvements.

Chapter 4

Implementation

Two applications were implemented to get a better insight into the mechanics of audience feedback scenarios on students' smartphones. They were used to prove the validity of the models described in Chapter 3 and to gain experience for further improvements. The first prototype, the *MobileQuiz*, was written as a successor to the *WilMa*-system of Scheele et al. [177]. It was implemented according to the simple model as a pure audience feedback system. It is a native plug-in for the university's e-learning platform ILIAS and is still used today by dozens of lecturers.

After realizing that the basic model, and therefore the *MobileQuiz*, reached its limits and was not able to satisfy the growing needs of our lecturers, the other, more abstract model was designed and implemented as a new prototype: the *MobileQuiz2*. The *MobileQuiz2* was constructed to satisfy the new requirements of generality with customization, which were not fulfilled by the first *MobileQuiz*.

4.1 MobileQuiz

The *MobileQuiz* application is an audience response system, designed as a plug-in for the university's learning management platform ILIAS. It was implemented as two separate parts: a *backend* providing the view for the lecturer and a *frontend* for the participants of the quizzes. The lecturers' backend is used to create new questions and answers, to control the quiz rounds and to display the results. The students' view displays everything needed to interact with the quiz on the students' mobile devices. Whereas the lecturers' view was implemented to meet the plug-in conventions of the e-learning platform ILIAS, the students' view is a separate, self-containing application based on PHP and JQuery Mobile. Although the students' view is independent of ILIAS, it is delivered in the same software package. This simplifies the effort for the installation and maintenance of the system. Besides the same folder environment, the frontend view has no direct connection to the ILIAS application. The two parts communicate indirectly through the database. They share the same database tables. Figure 4.1 shows how both parts access data from the same database while being technically independent. The independence of the two parts simplifies eventual replacements at a later time.

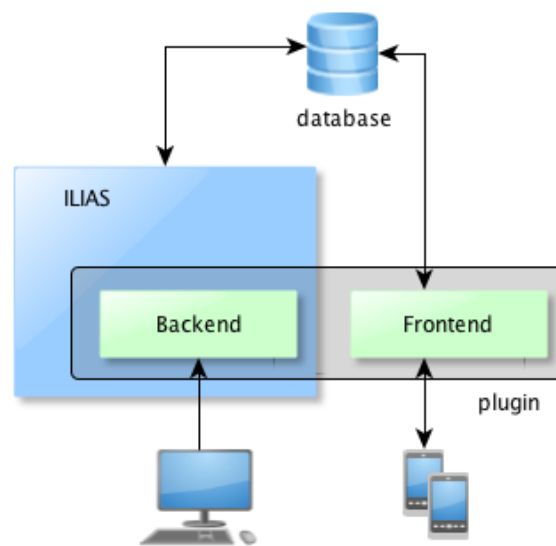


FIGURE 4.1: The *MobileQuiz* consists of a lecturers' view (backend) within ILIAS and a separate students' view (frontend).

4.1.1 Requirements

Before the *MobileQuiz* application was implemented, a set of requirements had to be defined (see Schön et al. [5]):

1. No additional software should be necessary.
2. Support of almost every modern mobile device.
3. Integration into our e-learning management platform ILIAS.
4. Access as quickly as possible, no complex authentication necessary.
5. Fast and easy creation of new quizzes.

Firstly: The most important requirement was that neither students nor lecturers had to install any additional software. It was more convenient to only require standard applications like a web browser. The computers within the university's lecture halls, the lecturers' laptops, smartphones, tablet PCs and similar student devices were all equipped with at least one of these.

Secondly: Students should be able to participate in the quizzes with every smart mobile device, independent of its brand or operating system.

Thirdly: At the time of the implementation, learning management systems had become very common at universities, and every lecturer at the University of Mannheim owned an account by default. For that reason, the integration of the lecturers' view into the existing learning management system was a main requirement to keep the efforts to enter and use the system as low as possible. Therefore, the *MobileQuiz* was implemented as a regular plug-in for ILIAS.

Fourthly: The access to the quizzes should happen fast and easy. No complex authentication should be necessary. It was decided to use a shortened URL and a QR code

for easy access were intended. This solution came with the sacrifice of a secure user authentication. Everybody in sight of the QR code would be able to join the quiz.

Fifthly: The creation of new quizzes was supposed to be simple and fast. In that way a lecturer should be able to create new quizzes by himself, and if necessary, ad-hoc during the ongoing lecture without losing much time.

At the time of the implementation, none of the available ARSs fulfilled all of these requirements.

4.1.2 Technology

The implementation of the *MobileQuiz* was mainly based on technologies used by systems, which were already used at the University of Mannheim. Additional libraries were common solutions for modern web applications at the time of the implementation. They are explained below.

ILIAS

One of the first design decisions was that the *MobileQuiz* application had to be a plug-in for our existing learning management platform ILIAS. This already specified most of the technologies as the plug-in had to implement the given interfaces. ILIAS is an open source application written in PHP. It offers an open interface structure which provides several build-in features for customized plug-ins like the user and authentication management. The *MobileQuiz* was implemented in the form of a *repository object*. *Repository objects* are elements which can be used within courses and groups and contain learning information. Already build-in objects are regular tests, wikis and forums. Lecturers were therefore able to create a quiz in the same way as they create a forum or upload a new file to their course. Usage of the ILIAS framework made the integration of the underlying MySQL database almost mandatory. The *MobileQuiz* could also make use of the existing, fine-grained roles and rights management and did not have to implement one for itself.

PHP

PHP is a script-based coding language for web applications. The main reason for its usage was due to ILIAS is implemented with it. Further benefits were that PHP was quite common and very widespread. It had a large community and therefore a reliable developing team. PHP applications usually run on an Apache HTTP Server, which is one of the most common and free web servers available.

MySQL Database

The plug-in had no specific requirements on the database used. Any common relational database management system would be sufficient. The already existing MySQL database of ILIAS was used to store all the application's data. The *MobileQuiz* plug-in added its own tables to the default ones of the ILIAS core application. These were filled by the

lecturers' view via the ILIAS database interfaces, and directly accessed by the students' view.

jQuery Mobile

jQuery Mobile is a HTML5-based library which provides user interfaces to create responsive web pages and mobile applications that are accessible on many smartphones, tablets and desktop devices. By the time of the implementation, the propagation of smartphones was still evolving. jQuery Mobile had the advantage to automatically scale the appearance on the device to the device's hardware. In this way, the fancy rounded elements displayed on modern smartphones were replaced by simple angular shaped elements on plain old handheld phones.

Charts

The first prototype of the *MobileQuiz* used self-implemented charts. These were based on CSS and HTML containers to create colorful representations of the summarized results. After a first revision, the more powerful Highcharts library was used for plotting the charts. The Highcharts library was well established at that time and offered a wide range of additional features. The charts looked more modern and acted more dynamically. Highcharts is free to use for non-commercial purposes.

QR Code

Quick Response Codes (QR codes) were used to simplify the students' access to the quizzes. They displayed the link to the current quiz in the form of a machine-readable image of black and white squares. Students could then use their devices' camera, together with a standard QR code reading software to access a quiz round. They did not need to manually type in a link on the tiny keyboards of their mobile phones. QR codes are two-dimensional barcodes storing the data in square patterns of black modules on a white background, as seen in the background of Figure 1.1. Although this technology can be used to encode any kind of data, it is especially useful to represent a link to a web page. QR codes are often found in advertisements, linking to further information about a product. There are several standards, including ISO/IEC18004 for the physical encoding, and a de-facto standard for encoding URLs from NTT DoCoMo. This standard is optimized for fast readability and includes error correction to enhance its robustness. The URL was shortened with the university's own URL shortening service before it was encoded within the *MobileQuiz*. This further improved the readability as fewer characters enlarge the size of the single squares.

4.1.3 Database Schema

Because the lecturers' view was implemented as an ILIAS repository object, it also used the ILIAS database and its naming conventions for tables and columns. The name of a table usually contains the expression *rep_robj* (used for all repository object plug-ins) followed by the plug-in's unique name and a specific table name. The unique name of

the *MobileQuiz* is *xuiz*. ILIAS uses PEAR's MDB2 as a database abstraction layer and supports MySQL and Oracle databases. MySQL uses a specific auto increment which ILIAS replaced with the MDB2's technique called Sequences. Sequences are separate database tables to every usual database table that increment a special sequence field to generate unique values.

Figure 4.2 displays an ER Diagram of the *MobileQuiz*'s data model. The table *quizzes* represents the different quiz repository objects. Every quiz can contain several *questions* which in turn contain several answer *choices*. Additionally, a quiz can have several *rounds*. During each *round*, students submit their *answers* according to the available *choices*.

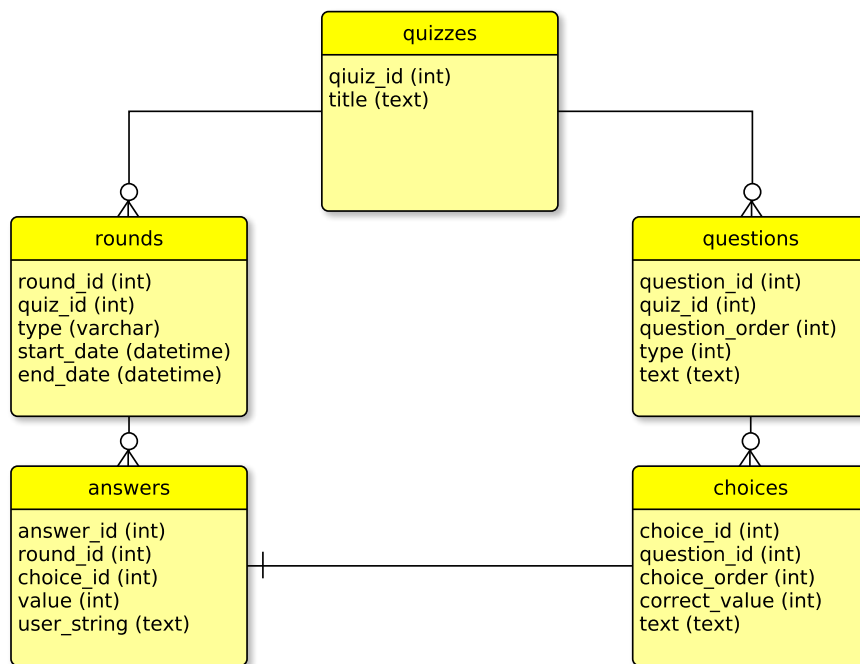


FIGURE 4.2: ER Diagram of the *MobileQuiz*'s database schema.

Quiz Table: The *quizzes* table (`rep_robj_xuiz_quizzes`) represents the quizzes. The information about the quizzes is additionally stored within the ILIAS tables where the common information of repository objects is stored. The *quiz_id* is equal to the *object_id* assigned by ILIAS, and the quiz's title is similar to the object's title. Despite this redundancy, the table is important to provide the student's view with this data, as it cannot access the ILIAS core tables.

Question Table: The *questions* table (`rep_robj_xuiz_questions`) contains the questions belonging to a specific quiz. It stores the question's type and the question's text. The *MobileQuiz* supports three kinds of question types. *Single Choice* questions with one answer possibility, *multiple choice* questions where many answer choices can be checked and *numeric slider* questions where a numeric slider enables the input of a number between a given minimum and maximum. The slider questions also have a step range, a correct answer and an answer tolerance. The column *question_order* contains

a numeric value which is used to order the questions when displayed. The lecturers can change this order during the creation process of the quizzes.

Choices Table: The *choices* table (`rep_robj_xuiz_choices`) contains the choices which belong to a specific question. Questions can have multiple choices. A choice is usually displayed as a possible answer to the students. The column *correct_value* is used to store a *true* or *false* value to indicate if the possible answer is correct. This information is later needed to calculate the student's result and colorize the result charts. If the question is of the type *numeric slider*, the field contains the minimum, maximum, step width, correct value and answer tolerance as JSON values. Similar to the *question_order* the *choice_order* contains a numeric value, which defines the displayed order of the choices in a question.

Rounds Table: The *rounds* table (`rep_robj_xuiz_rounds`) contains the actual quiz executions in the lecture. Every time a new quiz is performed an entry for this table is created, and the current time of start is stored. The end time is written as soon as the round is stopped. The column *type* stores the flag for the *HomeQuiz* feature. With a positive flag, the quiz can be repeated several times, and the individual results are directly displayed after the student submitted his answers.

Answers Table: The *answers* table (`rep_robj_xuiz_answers`) is the only table which is solely filled by the student's view. Whenever a student submits an answer on its mobile device, it is stored in this table. The lecturer's view then reads and summarizes these answers to create the result charts. If the student selected a given choice, the value of the answer is represented by the number '1'; it contains a '0' if it was not chosen. In case of a *numeric-slider* question, the number submitted as an answer is stored. The *user string* field is filled with a GUID by the student's view and marks all answers of one student.

4.1.4 Students' View

One of the *MobileQuiz*'s main goals was to make the application compatible with as many devices as possible. It should be easy for the students to connect to the *MobileQuiz* and to answer the questions. Furthermore, the application had to be robust regarding connection failures and a slow WiFi connection. As mentioned, the quiz was implemented as a web application by using existing web technologies based on the jQuery Mobile web framework. A main advantage of this approach was the high accessibility for a wide range of mobile devices, including smartphones, laptops, netbooks and tablet PCs, as well as stationary devices like desktop PCs and workstations.

Furthermore, these web technologies opened up further opportunities. Although the quiz appeared as a native application for smartphones, the questions were presented as usual web pages, opened by the devices' browser. Therefore, it was quite easy to include pictures, videos, and further media content, as they are usual web elements.

Figure 4.3 shows a question related to an included picture. The picture was not directly uploaded to the application, but referenced from a Wikipedia page. The same concept

enabled the embedding of more complex types of multimedia content like videos and HTML gadgets. Explanatory videos could be included from e.g., Youtube in the same way. They were not uploaded directly to the applications destination, but could be referenced from the quiz. Unfortunately, mobile browsers usually do not support the latest web technologies to the extent desktop browsers do. When HTML5 3D technology was applied in the quizzes, only a few students were able to watch the content flawlessly. Only web multimedia technologies, that are already widely available for mobile browsers were used in regular quizzes. These were especially pictures and videos.

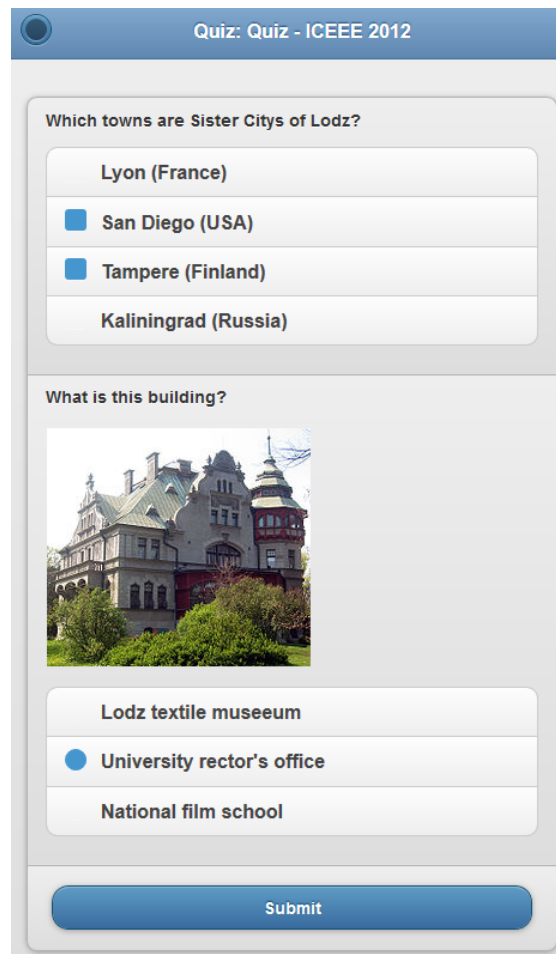


FIGURE 4.3: Students' view of the *MobileQuiz*.

The actual code for the students' view was structured as a small self-contained PHP web application. It followed the MVC (Model View Controller) pattern and consisted of less than 30 PHP files altogether.

4.1.5 Lecturers' View

ILIAS is the university's only e-learning platform, which addresses all lecturers, students and the majority of courses. One of its components is already a survey tool to support many different question types. Unfortunately, it does not meet the requirements stated above very well, because it was not designed to create quizzes for the students' mobile devices within a short time, e.g., during a lecture. Furthermore, it does not allow anonymous quizzes as the students have to be logged into the system in order to participate.

However, being a productive system, it already provides rights and role management, and most of the lecturers are familiar with the application. Thus the quiz's administration part was developed as a plug-in for ILIAS. All the management activities, like creating questions, starting quiz rounds and the visualization of the results can thus be done within ILIAS.

Following the ILIAS plug-in interface the lecturers' view consists of ten PHP classes and nine HTML view files. The file names follow the ILIAS plug-in guidelines. As an ILIAS repository object, the lecturers' view was fully integrated into the existing layout, as depicted in Figure 4.4. It allowed the lecturers to create and edit quizzes, to open new quiz rounds and to display the result charts. The QR code and URL to the quiz rounds could be displayed in a maximized format, to display them on a screen.

ILIAS

UNIVERSITÄT MANNHEIM


PERSONAL DESKTOP ▾ PORTAL² REPOSITORY ▾ ADMINISTRATION ▾ HELP

Repository » Einrichtungen » Hochschule » Universität Mannheim » Hochschulleitung und Organe » MultiMediaZentrum » MobileQuiz » Quiz - ICEEE 2012

Quiz - ICEEE 2012

Current quiz round Questions & Answers Rounds & Result Properties Permissions Further Information

QUESTIONS
(1 - 2 of 2)

Question	Type	Choices	Edit	Delete	Up	Down
Which towns are Sister Cities of Lodz?	Multiple Choice	4	edit	delete	▲	▼
What is this building? 	Single Choice	3	edit	delete	▲	▼

(1 - 2 of 2)

powered by ILIAS (v5.0.0 2015-02-06) | Legal Notice | Contact Administration

FIGURE 4.4: Lecturers'view of a quiz embedded into ILIAS.

4.1.6 HomeQuiz Feature

Besides the usage in the classroom, when the QR code was displayed on the front screen, the *MobileQuiz* could also be used in a homework environment, as published in Schön et al. [8]. The goal was to transfer the benefits of using an ARS within a classroom to the students' preparation and rework phase at home and thus bypass the drawback of loosing much time during the lecture. For that reason, classical paper learning materials were blended with modern digital self-assessment tests. The well-accepted *MobileQuiz* was enhanced with a self-assessment functionality. The resulting *HomeQuiz* feature

used the well-functioning QR codes and printed them on paper sheets. The codes then were a link to a content-related digital test. The approach was to print these codes on learning materials, like exercise sheets and lecture scripts as shown in Figure 4.5. Students then invoked the quizzes by using their own web-enabled mobile devices. The QR code allowed them to easily switch from the paper script to the digital test.

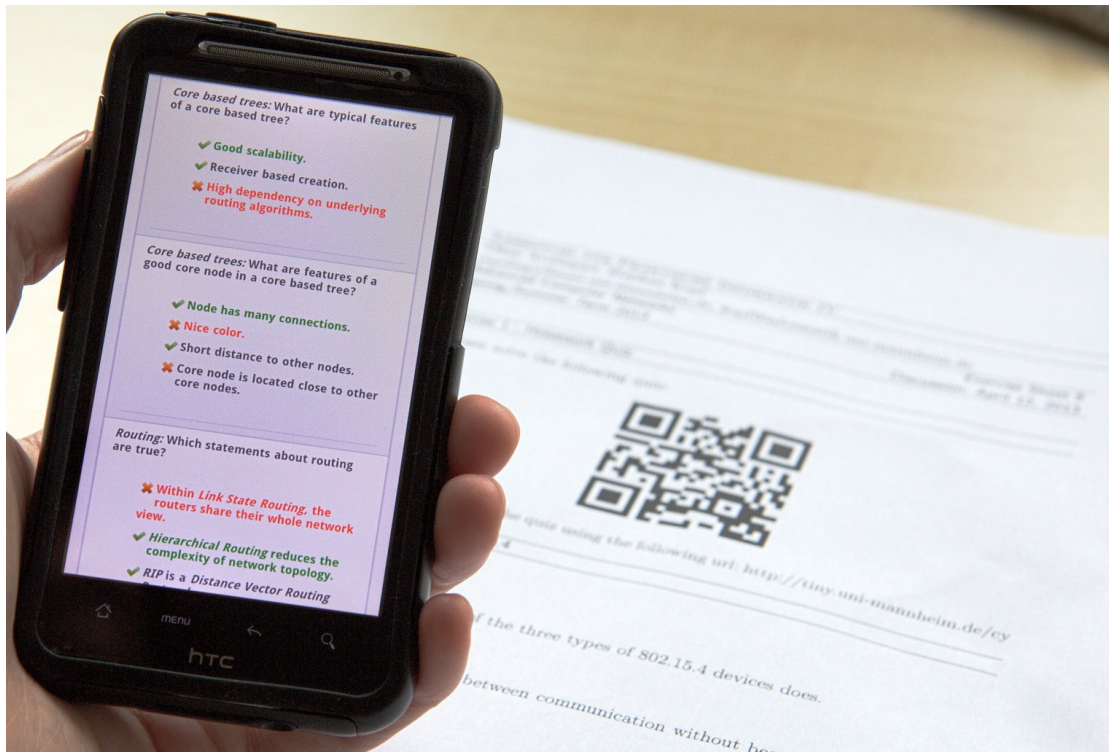


FIGURE 4.5: QR code printed on an exercise sheet for self-assessment.

The quiz rounds could therefore be set to the status *direct feedback*, which changed some of the usual application's behaviors. The one time restriction of the frequency of participation was disabled. So, every student could repeat the quiz as often as wished. Additionally, the students' individual results of the current quiz were displayed to the student after the answers were submitted. Therefore, the quiz could be used more efficiently in self-learning settings at home or in preparation of an exams.

The submitted results were also aggregated and displayed, as in the usual *MobileQuiz* application before. Using these results, the lecturer could get an impression of the students' knowledge before the next lecture started, and could prepare it accordingly.

4.2 MobileQuiz2

After realizing that the first version of the *MobileQuiz* was not able to fully satisfy the lecturers' needs, the new model, presented in Chapter 3, was created to describe a large number of possible, individual learning scenarios. To proof the validity, the power and the weaknesses of this approach, an additional application, the *MobileQuiz2* was designed to implement this more advanced, abstract model. The new prototype should adopt the technologies and functionalities of the basic *MobileQuiz*, which were approved

to be useful, and change these which they were not. It also should be based on the current state of the art technologies.

4.2.1 Requirements

Implementing the first *MobileQuiz* as a web application appeared to be a very good decision as the tool was easily adopted by the students. There have been very few problems with the different smartphone brands and versions. Therefore, the successor application was designed as a web application, too. The experiences with the QR code were equally positive. The students had only minor problems to access the quizzes by scanning the code from the projected screen. The shortened URL enabled a fall back access for those who were not able to read the QR code or tried to access the quiz with a device without a camera, e.g., a laptop. The combination of QR code and shortened URL was therefore used again.

The integration into ILIAS was not evaluated so positively. On one side, the integration into the ILIAS's rights and roles management and the fact that every lecturer is a user of the university's system by default lowered the effort of entrance. Each of our lecturers was able to easily access a quiz without installing new software and without learning any new application. On the other side, the deep integration into ILIAS combined with the URL shortener made the *MobileQuiz* very specialized to the environment of our university. Other universities were often not able to adapt the software equally as they were not using ILIAS. So, despite being an official ILIAS plug-in the usage of the *MobileQuiz* was mainly restricted to the University of Mannheim. The next version therefore had to be a standalone application. To realize something similar to an ILIAS integration, the new application had to offer an interface, which enable an ILIAS plug-in to connect to. By this, the main application would be independent of the individual ILIAS distributions, and other learning management systems would be supported in a similar way. The new application should offer the possibility to create and perform quizzes in a simple way, like the *MobileQuiz* did. But even when the complexity of the scenario rose, the lecturers should always be able to work within the same editors.

As a new requirement, the novel application should handle the different scenarios in similar ways. Layout and appearance should automatically be adapted according to the scenario's blueprint to reduce the complexity for users as much as possible. In addition to that, a main requirement of the new *MobileQuiz2* was to implement the model described in chapter Chapter 3 and to prove its validity. Further requirements, which were mandatory for a productive application, were considered, but not highly prioritized during the implementation of the prototype. Examples are the performance of complex and interactive scenarios in lectures with more than a hundred students, interfaces to other learning platforms and the transferability of quizzes into or from other audience response systems.

4.2.2 Technology

Zend Framework

Like the *MobileQuiz* and ILIAS, the *MobileQuiz2* was coded in PHP. PHP is easy to install and run on Apache HTTP Servers, which is one of the most common web servers

in the World Wide Web. The *MobileQuiz2* prototype application was implemented in the Zend framework with its current version 2. Zend is a powerful environment for modular, object-oriented web development in PHP. The main part of the application with all its server-side business logic was written with this framework.

Zend uses modules to encapsulate the logic parts. As seen in Figure 4.6, the application is structured in four main folders, *config*, *data*, *module* and *public*. *Config* stores all configuration data, like the connection to the database and references to the file system. *Data* holds uploaded and stored files of the application. Uploaded scenario blueprints are stored here, together with the application's log files. The *module* folder contains the applications modules for business logic. Every area of the business logic is encapsulated within its own module folder and can therefore be replaced easily. The *public* folder contains all data directly accessible by the clients' browser. Images which are displayed on the web page and QR codes are stored here. CSS files for the layout, JavaScript files for the surface and the used libraries (e.g., JQuery, JqPlot) are also stored in the *public* folder.

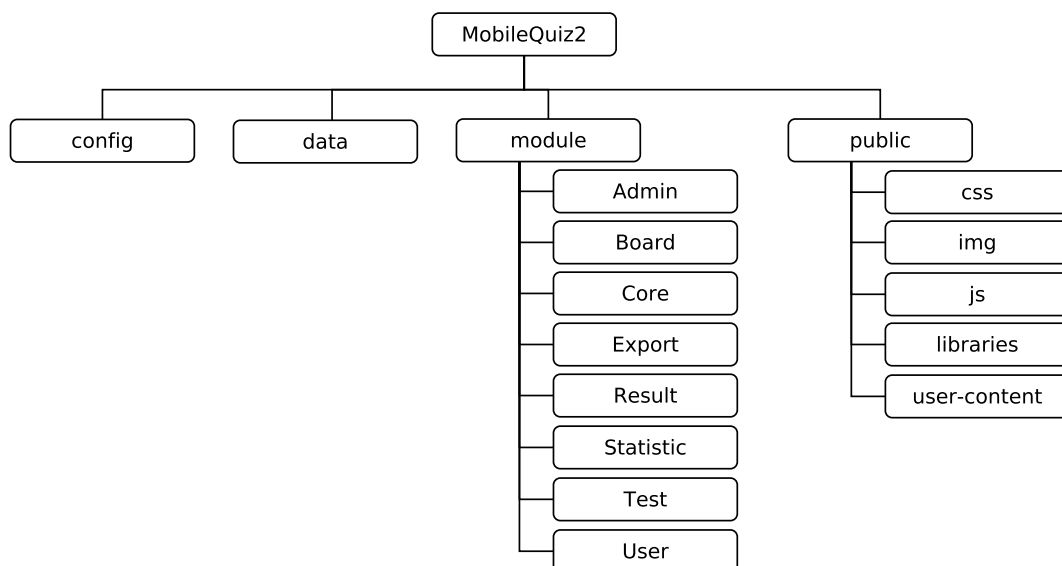


FIGURE 4.6: The structure of modules and classes within the *MobileQuiz2*.

AJAX and Web Technologies

Modern web technologies like AJAX, HTML5, CSS3 and the JavaScript-based web frameworks JQuery and JQuery Mobile were used for the visualization and direct user interaction. HTML5 and CSS3 offer modern ways of integrating media elements and the page's style. AJAX enables steady, asynchronous connections between core application and clients on the students' devices. Therefore, it is possible to trigger events from the server side and inform the clients to react in response.

Charts

Dynamic and appealing charts are realizable with the JavaScript-based library JqPlot. JqPlot offers a variety of different chart types, which can be plotted dynamically on the clients browser and are easy to customize.

4.2.3 Database

The *MobileQuiz2*'s database schema is equal to the data model in Section 3.4.3. Figure 4.7 shows an ER diagram of the database tables with their attributes and their relations to each other. The core application is based on the three phases: blueprint (scenario), quiz (entities), game rounds (games) with their respective objects and attributes.

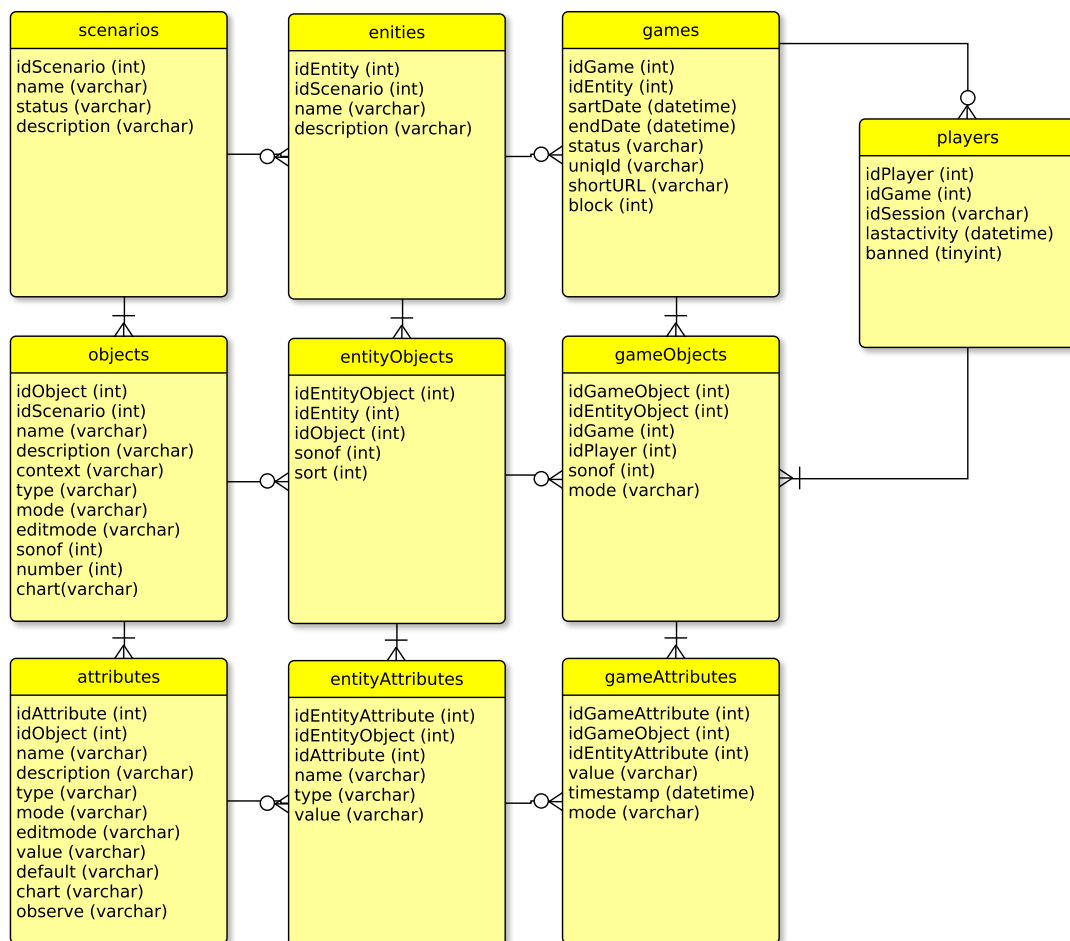


FIGURE 4.7: ER diagram of the database schema of the *MobileQuiz2*.

Scenarios: The scenario table stores the basic information. These are mainly the name of the scenario and a general description. The scenario's status marks if the scenario is *active* or *inactive*. Only *active* scenarios can be chosen to create new quizzes. *Inactive* scenarios can not be used to create new quizzes, but already existing quizzes can still be executed.

Objects: Objects store the information about the scenario's structure. Besides basic information, like the id of the corresponding scenario, a name and a description, they build the core structure as they can outline a tree via the *sonof* relation. Every object can be the son of another object. E.g., an answer object is a son of a question object. Objects also have a given context, which specifies if they are created only once when starting a new game or individually for every new player. *Mode* and *editmode* describe how the objects appear to the students on the board (mode) and to the lecturers in the quiz editor (editmode). The *number* specifies how many objects should be created by default, when creating a new quiz entity. *Chart* finally states which chart type should be used as the default on the result pages.

Attributes: Attributes describe the appearance of the elements. As objects, attributes store some basic information, like the id of the corresponding object, a name and a short description. The appearance is determined by the attribute's *type* and *visibility*. Depending on the type (e.g., numeric, char, text, button, slider, boolean), the attributes will be rendered as input fields, text areas, buttons or check boxes. The visibility is determined by the *mode*, as it specifies if these HTML elements should be hidden, write-able or grayed out. Similarly, *editmode* specifies the visibility for the lecturer in the quiz editor. The *value* specifies the value of the attribute, whereas *default* describes an optional default value which is used as a pre-selection if it is not overwritten within a later phase. *Chart* states how this kind of attribute should be used in charts. It can be the *title* of a chart (e.g., a question text) or a legend (e.g., an answer text). *Observe* finally tags if the lecturer should be able to observe this attribute in the game master view.

Entities: *Entities* are the actual quiz entities. The entities table has a relation to the scenario, a name and a description field.

EntityObjects: *EntityObjects* have a relation to their entity and object blueprint. Similar to the object, they build the tree structure of the quiz with a *sonof* id of their father entityObject. The only additional information compared to the scenario's object is *sort*. This describes the sort order of the entityObjects in the quiz editor and the students' view on the board. It therefore specifies the sequence of questions on the students' phones.

EntityAttributes: *EntityAttributes* also have a relation to their blueprint attribute and their entityObject. They also have a *value* that can overwrite the value of the scenario's attribute. E.g., if the scenario's attribute has an example question text, it is overwritten with an actual, content-specific question text. *Name* and *type* were used during the first versions but were deprecated, as the static information is already stored in the attribute.

Games: *Game* stores the particular information of a performed game round within the lecture. These are mainly the start and end date and the status about its repeatability. The *uniqId* stores an id which is used in the URL for this particular game round and for the QR code identification. The *shortURL* stores the shortened URL to this game

round which is used to generate the QR code. The *block* flag is used for performance issues. As some parts of the application's algorithm takes a lot of time, a the rules of a game can be blocked to prevent a second iteration of the algorithm on this game until the current run is finished.

GameObjects: Similar to *objects* and *entityObjects*, *gameObjects* build the structure of the game's objects via the *sonof* information. They also have a relation to their *game*, *entityObject* and *player*, as a *gameObject* can belong to an individual participant. They have a separate *mode*, as their visibility can change dynamically throughout a game round.

GameAttributes: Similar to *entityAttributes*, *gameAttributes* store their relations to their superior *gameObject* and to their corresponding *entityAttribute*. They also have a separate *mode*, as their visibility can change dynamically throughout a game round, too. The *value* stores the actual students' input during the quiz execution, whereas the *timestamp* stores the time of the value's last change.

Players: Besides the relational information about the game and the current session of a game's participant, the player table stores the time of the player's last activity. This is used to give the lecturer a better overview about the current activity of the round. Additionally, every player has a flag which marks if the player was banned from that game. Banned players are not able to further participate in that game round.

Rules: Every rule belongs to a specific scenario and contains a *name* and a *context*. The name is only used to simplify the creation of a scenario and debugging. The context defines the amount of attributes which are used for the checks and actions.

Checks: The *checks* table defines the conditions of the rules. They always have a relation to an attribute against which the condition should be checked, an *operator* which defines the kind of evaluation and a value against which the condition is validated. E. g., a condition could check if the numeric input of the student is greater than the 58. More operators are described in Section 3.4.4. The value may contain a concrete value or a reference to another attribute containing the value. If the value refers to another attribute, the id of this attribute is stored in the field with a prefixed '&'.

Actions: *Actions* are very similar to *checks*. They have a relation to an attribute, an operator and a value for the actual action. The operators are mainly mathematical expressions like addition and subtraction. A typical action is to increase a given attribute by the action's value.

4.2.4 Students' View

The *Students' View* is implemented as one of the modules of the *MobileQuiz2* application. According to the described model in chapter Chapter 3, the student's view is the

board of the game and the module is named equally. It is based on two web pages: a start page and the actual board. Along with them, the module contains a start controller class and a board controller class for handling the user actions. The module has one service class to manage the more complex business logic.

After opening the *URL* of the current game via the QR code or the displayed URL, the start page is displayed. It offers little functionality and displays only a general information text and a button to join the current round. But beneath the surface, it handles authentication and organizational matters in the background. The page recognizes if the student already visited this game round and offers him the possibility to rejoin. If the lecturer allows students to restart their game, a second option is given to them: They can restart their game and enter as a new participant. Furthermore, if the student was banned from a specific game round due to trolling or improper behavior, the access to that round is blocked here. The page has additional, more technical functionalities. When the students entered the game round directly, it was observed, that the system generated almost twice as many virtual participants as really joined. Half of the participants showed no activity at all and only congested the database. A reasonable explanation for this was that the students' QR code scanners or browsers used a kind of proxy to check if the page at the given address really existed. They then gave it to the user and let him join again. So the page had almost twice the number of initial requests as real participants. Since then, a new user session is generated when a real user actively presses the button to join the current game round.

After this manual entering, the main board page handles everything during the actual game round. It is the real *board* of the game and it is used for every game of every scenario. Depending on the elements, which were defined in the blueprint of the scenario, the visible page can appear in a completely different look. Section 4.3 will describe different scenarios and their appearance. Among them is a test scenario with several multiple choice questions, and a game theory scenario which displays only a text, a slider and a submit button.

When entering the page, all game objects and attributes that are necessary for this student, are rendered. Objects and attributes which should not be seen by the student are hidden, but appear in the web page's Document Object Model (DOM). Visible elements are rendered as *readable*, or if the student should be able to enter an input, as *writable*. In this way, hiding and unhiding objects is technically easy and can be done with little CSS transformation and without refreshing the full page. The students stay on the same page for the whole game round, while the page dynamically changes its appearance by changing the visibility of its elements. An AJAX-request is sent to the server every five seconds and whenever the student makes a manual input. If triggered by a student's action, the request contains the input value, and the server stores and processes this change. The response contains all values relevant to this student, and the page is dynamically re-rendered, according to the received response. The values are updated, and modifications in the visibility of objects are done. Figure 4.8 explains this process.

The controller class `BoardController.php` receives every student's action, passes the information to the services, collects necessary data and forwards the students to the next view. Besides the first page call or error events, it is mostly addressed by an AJAX call and returns updated data back to the current board page. The controller checks the validity of the student's session and the status of the current game. It passes the

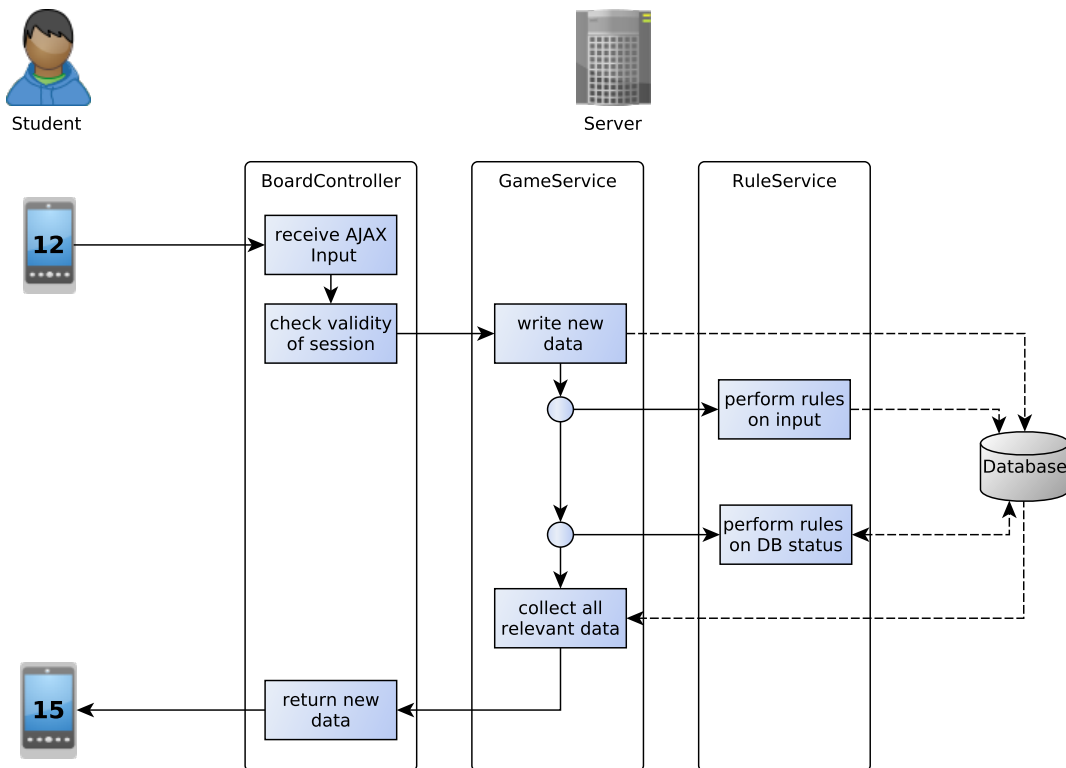


FIGURE 4.8: The process of the regular AJAX requests between the clients and the server.

modified data from the student to the core application, asks for the latest data from the database and returns it to the client. If any unexpected event occurs, the controller redirects the execution to a respective error page.

The service class `BoardService.php` offers a variety of helping service functions which are addressed by the controller. Most of these functions gather information about a given element, student or game. Others transform the collected information into a data structure which is used later to render the student's view.

4.2.5 Lecturers' View

The lecturers' view is far more complex than the students' view. The lecturers' view contains all administrative logic and pages as well as the actual heart of the application: the game round logic. The most important functionality, the implementation of the model of Chapter 3 is almost realized completely in the *Core* module. Other modules mainly contain functionalities which are necessary to operate the web application.

The Core Module

The *Core* module is the main part of the *MobileQuiz2* implementation. It is the largest and most complex module and implements the phases of blueprint creation and management (phase one), the creation and editing of the quiz entities (phase two) and the management of the game rounds (phase three).

Scenario Management: The *Scenario Management* is responsible for creating and managing the teaching scenarios. It contains the functionalities to upload a scenario blueprint in the form of a XML file, as shown in Listing 4.1. The listed scenario describes a Likert scale scenario, in which the lecturer can create several *DiscussionTopics*, which can then be rated by the students with a five point Likert scale slider. In addition to the functionality to upload scenario blueprints as XML files, a prototype of an online scenario editor has been implemented. It enables lecturers to create new teaching scenarios directly within the application.

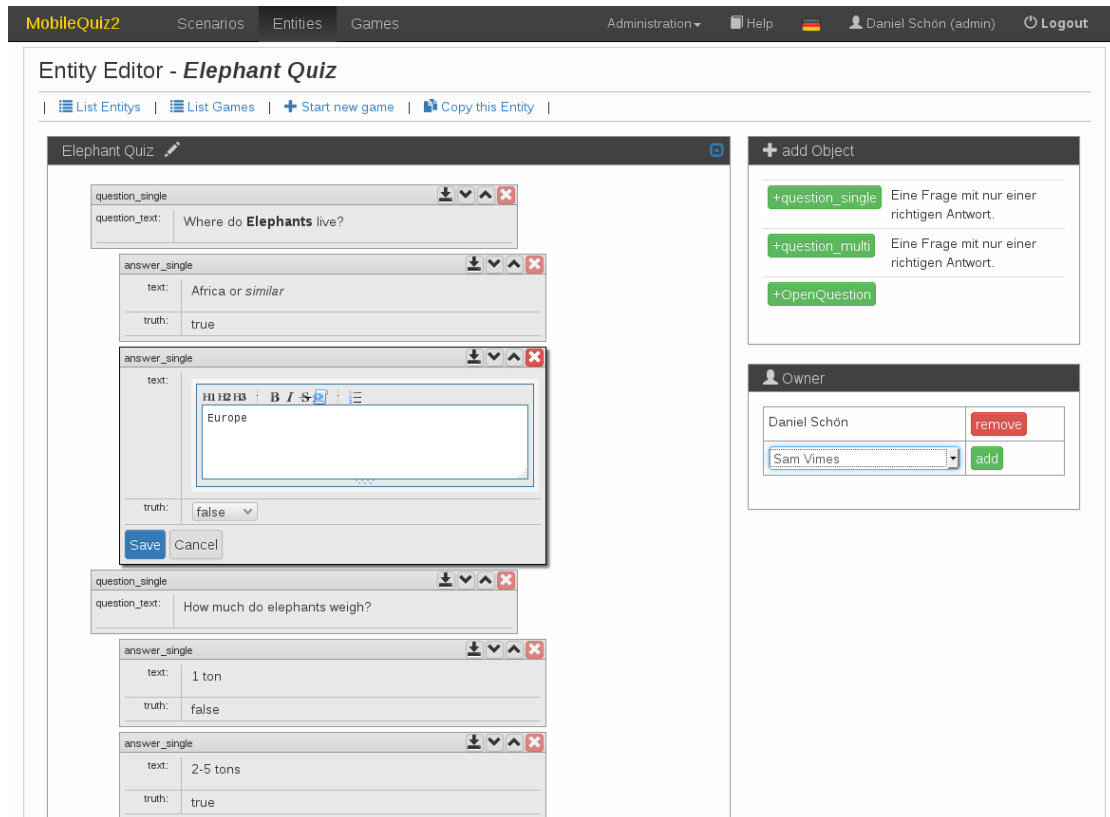
```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<scenario name="Likert_Scale_v02">
  <description>
    This scenario shows Likert Scale sliders from 1 to 5 with a description text.
    It has no submit button.
  </description>

  <objects>
    <!-- Slider -->
    <object name="DiscussionTopic" number="1" type="var" mode="write"
      editmode="write" context="player" chart="bar_vert">
      <attribute name="Question" type="text" editmode="write" mode="read"
        observe="false" chart="title"/>
      <attribute name="Valuation" type="slider" editmode="read" mode="write"
        observe="false" chart="data">
        <value>{"min":"1", "max":"5", "step":"1", "value":"3"}</value>
        <default>{"min":"1", "max":"5", "step":"1", "value":"3"}</default>
      </attribute>
    </object>
  </objects>

  <rules>
    <!-- no rules in this scenario -->
  </rules>
</scenario>
```

LISTING 4.1: XML file for a scenario blueprint description.

Entity Management: The *Entity Management* is responsible for the actual quiz creation. Beside the overview over the lecturer's quizzes it contains a powerful quiz editor. The editor automatically adapts the scenario's blueprint and offers the lecturer the possibility to create the quiz objects and enter the corresponding attributes. According to the blueprint's description the attributes are either readable, writable or not shown at all. Writable objects can be added, readable objects can be edited, and hidden objects are not displayed. Objects which belong to a given structure, such as quiz answers belonging to a specific quiz question, are visually arranged in an tree structure. Figure 4.9 shows the quiz editor for a *MobileQuiz*-like scenario. When using such scenarios, the editor should ideally look like the quiz creation page of the *MobileQuiz-ILIAS*-plugin.

FIGURE 4.9: Screenshot of the quiz editor of the *MobileQuiz2*.

Game Management: The *Game Management* is the heart of the *MobileQuiz2* implementation. It contains pages with an overview of current and closed games and the functionalities to open and manage new game rounds. Additionally, it offers a special *GameMaster* view for controlling the current classroom activity. It also provides the game logic.

The *GameMaster* view offers the lecturers an overview of current classroom activities. It has the functionality to start and stop opened games, display the QR code and links to the result page. The page also gives an overview about the current student participation. As students dislike unnecessary idle times in lectures (see Nielsen et al. [159]), this can help to frame game rounds to the right length. Lecturers recognize when no answering time is needed anymore and they can proceed with a discussion. Figure 4.10 shows a list of players, in which every participant has its own row. A serial number, the participants' idle time and the observed attributes are displayed. Observed attributes are defined in the blueprint of the scenario in the first phase. They may show the value of an attribute or an information if the students already submitted the entered solution. If the observed attribute is *true* or *false* the cell is automatically displayed with a green or red background. Thereby, the lecturers get a fast overview of the current activities. In Figure 4.10, the results of students' tasks of a "Question Phases" scenario are listed (see also Section 4.3.5). As every row shows one student, the green fields indicate the already submitted correct answers. The *GameMaster* page also offers an activity graph in the top right corner, which shows the participants' amount of actions. Every click made by a student counts as one action. When the activity decreases, the lecturer can assume that more and more students have finished the task.

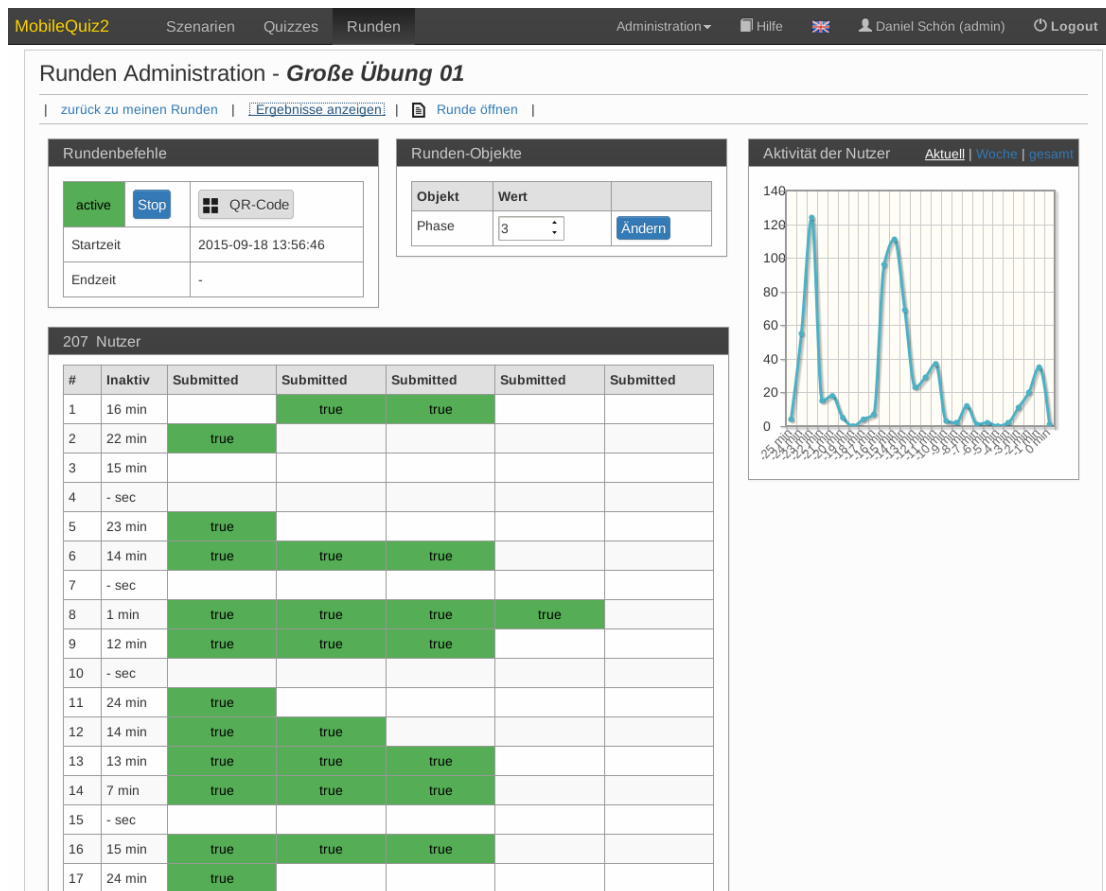
FIGURE 4.10: Screenshot of the *MobileQuiz2*'s Game Master view.

Figure 4.8 shows the `BoardController`, which passes the modified attributes to the `GameMaster`-service. The attribute's value is updated in the database, and the rules are triggered. After the rules are processed, the latest attribute values are requested from the database and returned to the `BoardController`, which sends them back to the client.

The scenario's game logic is mainly processed in the `RuleService.php` class. The service manages the execution of all rules defined in the current scenario. The class processes all rules in the order, in which they are defined in the blueprint. For every rule, the attributes of the defined context are taken and checked against the given conditions. If one rule is of context *input*, only the value of the current user action is checked. If the rule has a *player* context, all relevant attributes of this participant are checked.

The Result Module

The *Result* module draws all result charts of the information given by the students' inputs. The functionality was transferred into a separate module to simplify a possible exchange. A large variety of libraries is available to render charts, and the process of gathering, summarizing and preparing the data differs much between these libraries. In addition, even the different chart types within a single library require data in different forms. The chart library requires a different structured data array for every kind of chart. Twitter-like board messages are also displayed in a different kind than plain

text answers. Whereas plain text is listed in ascending order, dynamic messages are listed with the latest message first, together with a timestamp and with a functionality to hide comments or even ban the comment's author from this game round. This is necessary as students may be able to disturb the lecture with inappropriate contributions in scenarios in which they are allowed to write as many comments as they wish. The current implementation allows the lecturers to choose between several chart types, and it is able to update charts in real-time. The charts' colors are usually chosen dynamically. As a default, blue is used for every chart, but when counting or summarizing *true* or *false* values, the bars are automatically rendered in green or red.

The Export Module

The *Export* module is very small in comparison to other ones. It only offers some basic functionalities to return a downloadable CSV file for a given data array. It is used by other modules to offer a download functionality which can be opened by a spreadsheet application like *Microsoft Excel* or *Open Office Calc*. It is included into the view of every result chart and statistic view.

The Statistic Module

The *Statistic* module collects and analyses all kinds of data around the application's usage. It offers an interface to write statistical information into separate database tables. This interface is fed with information about the number of entities and games, as well as the timestamp of every user interaction. The amount of time a triggered user action needs until a response is visible on the student's device is further distinguished from the time the request crosses the network, the process time the system needs to perform the rules and several smaller steps within the application like gathering user relevant information from the database and rendering the view on the student's device. The data can be filtered and analyzed, and it can be displayed in several charts. Additionally, the raw data can be exported as a CSV file.

The User Module

The *User* module is necessary to manage the user accounts. Due to the standalone solution, the *MobileQuiz2* cannot use the user, rights and roles management of a superior framework, as the *MobileQuiz* does with ILIAS. Indeed, the *MobileQuiz2* uses some user management functionalities from the Zend2 framework. But due to its prototype status, it mainly implements a simple user and roles management solution for itself. The module offers the management of necessary user data like name, email, account and password. It further offers a standard login functionality and a simple rights management. Roles can be defined within the configuration file. These roles contain a set of rights which are verified before any user action is processed by the controller. The *MobileQuiz2* mainly uses three roles: A *lecturer* who can use predefined scenarios to create his own quizzes and manage his game rounds, a *specialist* who additionally has the rights to create new scenarios and the *administrator* with all available rights.

The Admin Module

The *Admin* module takes care of administrative functionalities. Among these are mostly consistency checks on the data within the database and the cleaning of obsolete information. In particular during the prototype's first test phases, data structures changed quite often, and inconsistencies happened. Verifying and cleaning the database from within the application simplified and accelerated the implementation process.

The Test Module

The test module offers several functionalities for testing the performance and database requirements of different scenarios. After we ran into several performance issues, the test module helped to simulate quizzes with a given number of participants and identified bottlenecks. A quiz of a given scenario can be chosen and the amount of related objects, attributes and rules are displayed. Then, a large number of simulated students can be generated into this quiz round. Together with these participants, all game objects and game attributes are created and stored in the database. On this data, the rules of the scenario can be performed and the required time be measured. Thereby, bottlenecks within the system can be identified easily and limits on the number of concurrent students per scenario can be set.

4.2.6 Scenario Editor

The advanced model of the *MobileQuiz2* allows a large amount of different teaching scenarios. In the first stage, these scenarios had to be defined in an XMLfile which needed some basic programming skills and knowledge in informatics. But only few of the lecturers who used the new application were able to create or customize new scenarios. To support the lecturers, a graphical scenario editor was designed and implemented with the cooperation of Thilo Weigold during his bachelor thesis [19]. The implementation and results presented in this section are based on his work.

The main objective of the graphical editor is to simplify the creation of teaching scenarios for the *MobileQuiz2*. The requirements and characteristics for the implementation were therefore oriented towards this main goal. Related authoring tools already described features which support the users in their task. Among these are suitability and functional capabilities for the specified purpose, the interoperability, re-usability of the created content and general usability of the tool. Focused on the *MobileQuiz2*, this means that the graphical scenario editor should provide an clear graphical interface for creating and customizing teaching scenarios. Lecturers should be able to create individual scenarios and modify the scenarios of their colleagues. The editor should further be directly embedded into the *MobileQuiz2* application and therefore implemented as a web application, which can be used with a common web browser. As the *MobileQuiz2* uses XML as the representation of a teaching scenario, the scenario editor had be able to read and export scenarios in this format. The authoring tool is supposed to facilitate the process of creating teaching scenarios by lecturers who have no programming expertise. Therefore it has to provide an intuitive environment for the construction and customization of new scenarios. Usability, which is mainly associated with learnability, was therefore a main consideration during the editor's conception phase. Especially for

non-technical users the tool had to be easy to learn and to use. Lecturers who use the *MobileQuiz2* shall be able to concentrate on the quiz itself, instead of being hindered by confusion and annoying technical issues. In addition, the scenario editor should provide the flexibility for lecturers to create and configure their individual scenarios. The degree of flexibility during the blueprint phase should meet the possibilities of the *MobileQuiz2*'s generic data model. But the balance between usability and flexibility to design complex scenarios is a limitation of general authoring tools. The *MobileQuiz2*, as well as the scenario editor, will be used by non-technical university lecturers and didactic experts. Therefore, full freedom and flexibility is required together with features, which improve simplification and assistance. Since the authoring tool is dependent on the functionalities of the *MobileQuiz2*, it has to support future extensions of functionalities and parameters.

When creating a scenario, the user ideally begins with a didactic concept and teaching goals and adapts the scenario's structure to them. Within the *MobileQuiz2*, a scenario is defined as consisting of five basic components: objects, attributes, rules, checks and actions. The scenario editor therefore has to support the lecturer in combining and defining these components in a structure that fits the intended purpose. But the editor only defines the blueprint, the concrete content of the scenarios is created in a later phase within the main *MobileQuiz2* application. The goal of the editor is to derive an unambiguous and abstract *MobileQuiz2*-internal representation of a teaching scenario. This should be achieved by giving the option to freely add, edit and delete components, and manually configure every property of each element. To support this creation of components, the editor provides the possibility to add pre-defined components. These pre-defined components are objects and rules, e. g., standard multiple-choice questions and submit buttons, which can be used in a large variety of different scenarios. Therefore, non-technical lecturers can simply create a standard scenario without the need to configure every detail from scratch. But expert users are still able to fully customize their scenarios.

As an example, the construction of a multiple-choice scenario is described in more detail. This scenario should contain questions, answers and a submit button. A question object with an attribute for the question's text is needed. The answer object has to be related to the question object, it also contains an attribute for the answer's text, an attribute for the correctness of this answer and a field for the player's input. All attributes have to be specified with the correct type. The object for the submit button needs an attribute of the type 'button'. Additionally, at least one rule has to be defined that reacts on the input and informs the player.

It is important to keep the frequency of user errors low, right from the beginning of the scenario's life cycle. Mechanisms which prevent inexperienced users and experts from implementing typical errors are offered by the application. These validation mechanisms can be divided in two general types. The first prevents errors directly during the authoring process, the other verifies a scenario's integrity before releasing it to further phases. The correct configuration of a component's properties can be checked during the creation process and directly signaled to the user. When defining a new check or action, it has to be assigned to an object or attribute, therefore a restricted selection of meaningful attributes is provided. The same method is used for the choice of a father object in the *sonof* property. More complex validations are also checked before approving the scenario for the next phase.

As the creation of teaching scenarios can be quite complex, the application provides an intuitive user interface. Figure 4.11 shows the scenario editor implemented into the *MobileQuiz2* application. The interface only works with one scenario at once. The screen is divided into a few areas of concern, defined by the underlying concept. At the top, an overview with all of the specific data is shown, including name, description and status. An action panel with the functionalities to add new components and predefined scenario components is provided beneath. Further, an overview of the scenario's current structure is provided in the areas 'Scenario Structure' and 'Rules and Interactions'. The section 'Scenario Structure' lists all objects in dark blue and assigned attributes in green. Rules, checks and actions are displayed in the area 'Rules and Interactions'. Element-specific actions and data are directly integrated into the respective UI element. When clicking an object, a form opens and displays the object's properties. The graphic representation of a scenario is based to the underlying XML structure. This represents a solution which does not limit the requirements and supports the creation of complex scenarios.

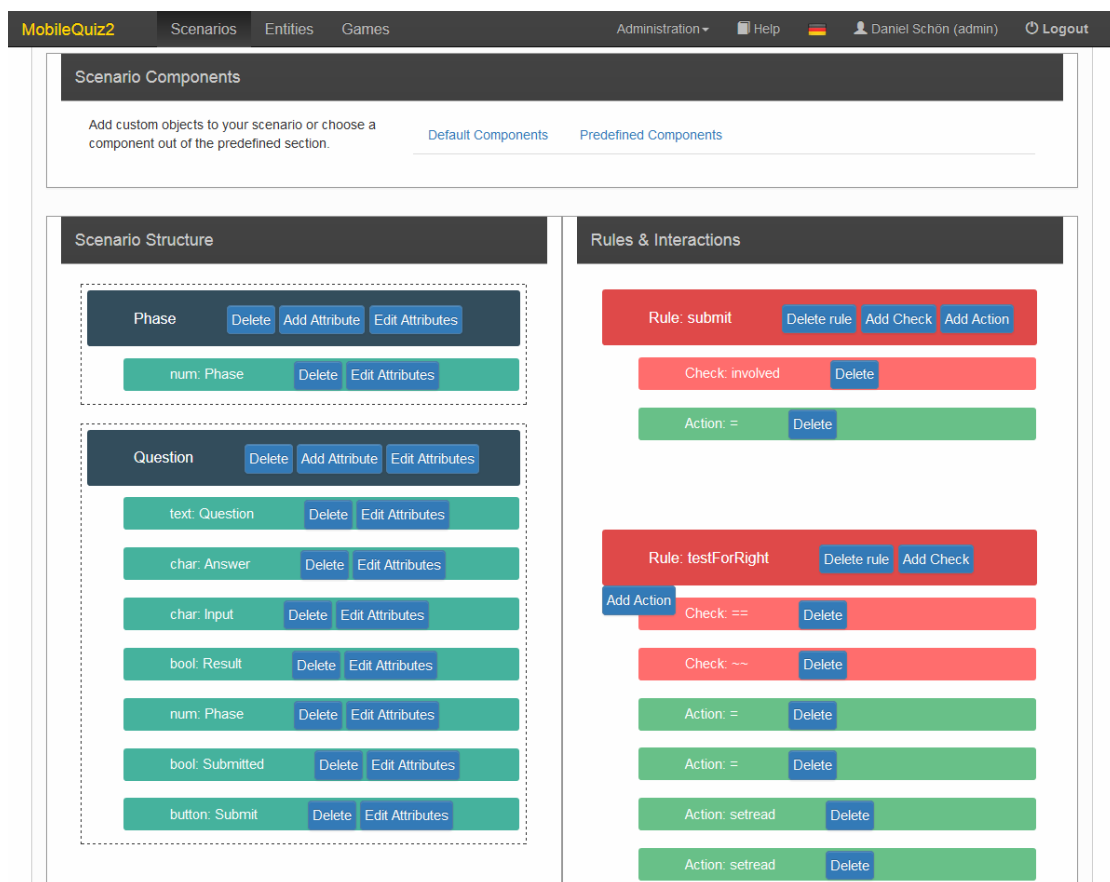


FIGURE 4.11: Screenshot of the *MobileQuiz2*'s graphical scenario editor.

The prototype of the scenario editor is a completely integrated application within the *MobileQuiz2*, which is implemented in PHP and uses the Zend framework version 2. The scenario editor is implemented as separate module in the overall application. It uses the same database as the main *MobileQuiz2* and does not require separate tables. The main part is implemented as a JavaScript application and is responsible for displaying the user interface and for simple display logic. The AngularJS2 framework is used as it is particularly suitable for this purpose. AngularJS especially supports the development of

interactive single-page applications. The scenario editor is based on two parts: an *index file* and the *FrontController*. The index file is the template which defines the overall structure of the user interface in HTML and is initially rendered on the server side. The *FrontController* is responsible to manage the view of the AngularJS application by synchronizing the application's view and the DOM elements behind it. If the scenario or one of its components is edited, the respective DOM elements are updated. The *FrontController* updates the data of the view by assigning data to the model. Several models were defined, including models for the scenario and each of its components. The integration of directives into HTML extends the DOM elements with functionalities and binds specific model data to them. The technology of this two-way data binding is used throughout the editor. The synchronization between DOM and view model proved to be very beneficial and useful.

4.3 Examples of Use

Due to the high flexibility and adaptability of the new model, it can be used for a large variety of different scenarios. Because of the new flexibility, the novel *MobileQuiz2* supports a larger variety of teaching scenarios, which were supported by specialized applications before. But the *MobileQuiz2* also supports the basic scenarios of the original *MobileQuiz* and additionally enables the customization and combinations of these. The lecturers at the University of Mannheim currently use the *MobileQuiz* and *MobileQuiz2* in different situations. This section gives a short overview of various applications.

4.3.1 Single and Multiple Choice Questions

The most popular scenarios mainly use only single-choice and multiple-choice questions. They are typically used to give the lecturers feedback on the students' current knowledge. Lecturers usually create about two to six questions with corresponding answer possibilities concerning a course-related topic. Students read these questions on their mobile devices, choose the correct answer(s) and submit their choices. Figure 4.12 shows a screenshot of a multiple-choice question scenario in a computer networks lecture. The submitted answers are aggregated, and the result can be displayed in front of the classroom. Usually, the result is then discussed with the students. The main benefit of this scenario is that the students are forced to reflect their own knowledge of the content, and the lecturer gets a realistic impression of the students' knowledge gaps.

Besides knowledge questions during the lecture, the scenario can also be used outside of the campus environment, like the *HomeQuiz* feature of the basic *MobileQuiz*. When printing the QR code on a paper exercise sheet, students can answer the questions in preparation of an upcoming lecture. The lecturer is then able to prepare the lecture according to the result of this pretest. In addition to questions about the course's content, single or multiple choice questions can also be used in surveys. E.g., a lecturer of 'Praktische Informatik I' used this scenario in the very first lecture of the semester to get an overview of the programming languages the students already knew.

Computer Networks - Quiz#04

What is the maximum number of bits for which we can correct 1-bit errors with a redundancy of four bits?

☐ 7

☒ 9

☐ 11

☐ 13

A parity bit (even parity) is used on a data transmission channel with 8 bit characters. Select all data packets where the parity check found no error.

☐ 011010111

☒ 101110010

☒ 000011110

☐ 101111001

FIGURE 4.12: Screenshot of a scenario with single and multiple choice questions.

4.3.2 Likert Scale Evaluation

Slider elements, as seen in Figure 4.13, can be used to create a Likert scale evaluation scenario. A lecturer of a weekend seminar in economic education during the Spring semester 2015 used such a scenario to let the students evaluate each others' oral presentations. The scenario consisted of several discussion points defined by the lecturer. These discussion points contained a textual phrase and a slider for the numbers one to five. The students opened the scenario on their devices with all discussion points, as seen in Figure 4.13, and were able to express their opinions continuously throughout the lecture. The results were aggregated and presented. No particular submission was necessary. The lecturer saw the students' opinions and recognized changes immediately. The summarized results were then used as a basis for follow-up discussions.

4.3.3 Lecture Feedback

Similar to the Likert scale scenario, the lecture feedback scenario uses sliders for collecting the students' opinion about the current lecture. Usually, this scenario is used to give the lecturer a live feedback about talking speed and comprehension of his or her presentation. As a further element, the scenario contains a text field to enable the students to give free comments. The scenario is used without a submit button, as the feedback is updated continuously. But this scenario is not used very often. Besides didactic annotations whether students are able to give a realistic feedback about their level of comprehension during the lecture, it is quite difficult for the lecturer to decide how to react to the given feedback. Assuming a situation in which five out of fifty students claim that the speed of progress is too slow, the lecturer has to decide if that means that

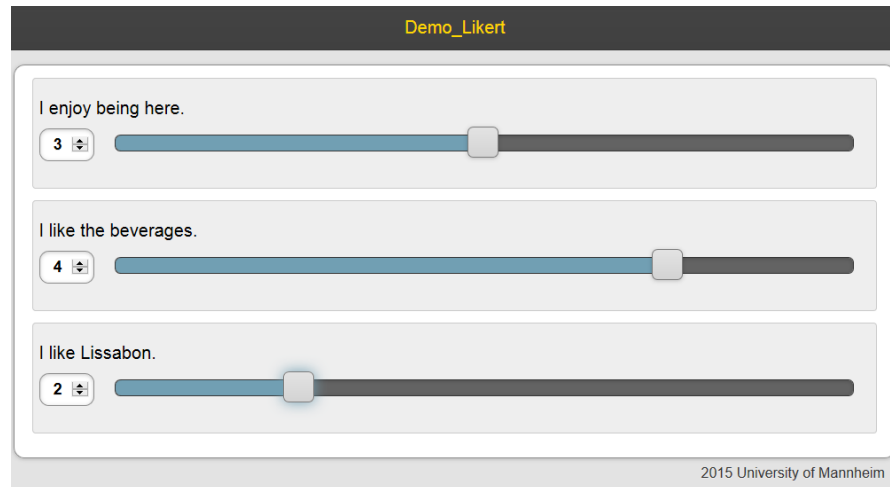


FIGURE 4.13: Screenshot of a scenario with Likert scale discussion points.

the remaining 45 students agree on the current speed of progress or that the remaining 45 students agree with the opinion of the five.

4.3.4 Message Board

Another specialized scenario for real-time comments is the message board. Thereby, the view for the students is limited to a short textual description, a text input field and a submit button. The students are able to enter a comment into the text input field, which is cleared again as soon as the comment is sent to the server by clicking the submit button. The comments are put on a Twitter¹-like message board, where the latest post is listed on top of the earlier messages. Figure 4.14 shows the message board with two highlighted comments from a large exercise of 'Praktische Informatik I' in the Fall semester 2015. The lecture was attended by about eighty students. The lecturer explained the current topics while he was programming the solutions of the students' exercises. He wanted the students to give a steady feedback and to ask current questions directly via the message board parallel to his programming effort. After he coded a snippet, he evaluated the comments and displayed them to the classroom. Within the first ten minutes of this lecture, he got about thirty comments and was able to discuss this feedback immediately with the students.

Unfortunately, after half of the lecture's time, some students got bored and started monkey business with the message board. They submitted nonsense and inappropriate comments. Therefore, the *MobileQuiz2* was improved so that the lecturer is now able to block inappropriate comments, highlight noteworthy ones, and is also able to ban students from active rounds as soon as they begin to disturb the lecture.

4.3.5 Question Phases

This scenario enables lecturers to go through several phases containing different tasks. Every phase is activated separately and can contain various types of objects. The

¹A mikroblogging service. <http://twitter.com>

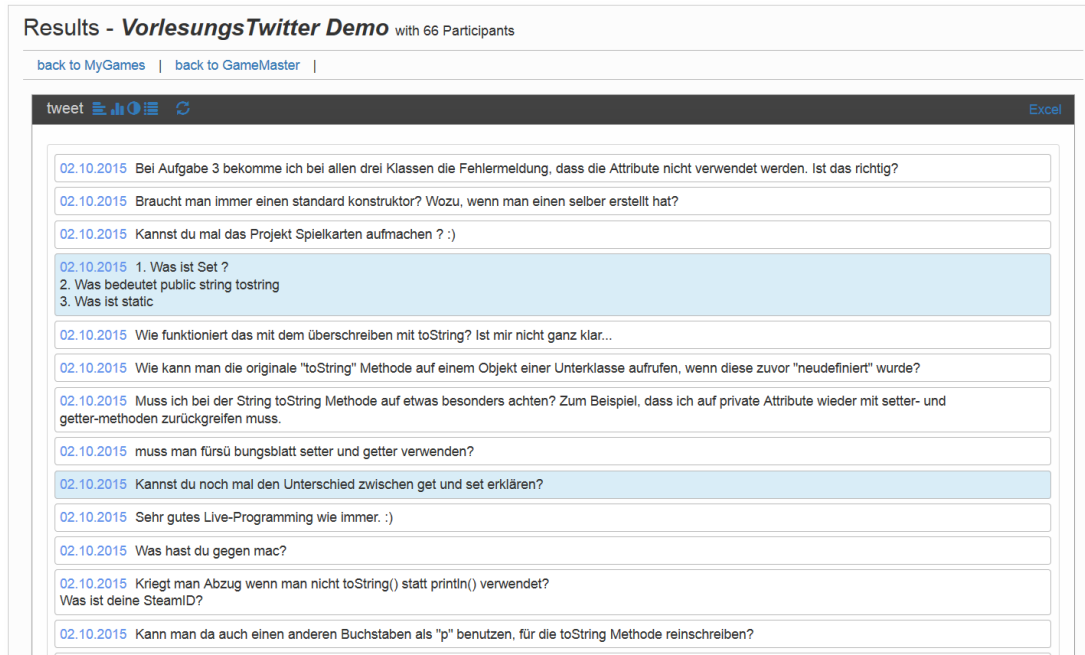


FIGURE 4.14: Listed comments of a Message Board scenario with highlighted comments.

scenario was used several times in slightly different variations. In the most complex variation, the lecturer of 'Praktische Informatik I' taught number conversion of binary numbers into the decimal and hexadecimal systems. He started the quiz in the first phase, and the students got the task to convert a given decimal number into the binary system, to enter the result into a text field, and to submit. The inputs were automatically compared to the correct solution, which was set by the lecturer during the quiz preparation (phase 2). The students got a direct feedback on their answers and got the possibility for a second try if their solution was not correct. Thereby, they were able to practice the conversion by themselves. The lecturer saw a list of submitted results on the *MobileQuiz2*'s **GameMaster** view and recognized the number of students who already mastered the conversion. After a discussion about the correct results and the procedure of the solution, he introduced the students to a more complex conversion and shifted the quiz into the second phase. The visible task on the students' devices changed automatically and they were challenged with a slightly more difficult conversion task. In this way, the students were guided through four phases and had the opportunity to exercise the number conversion for themselves with a steadily growing difficulty.

4.3.6 Guess Two Thirds of the Average

Beside the already mentioned scenarios, it is also possible to realize dynamic live experiments in the classroom. E.g., a professor of business economics used a game theory example called *guess two thirds of the average*. Students had to guess a number between zero and one hundred via a plain interface, as it can be seen in Figure 4.15. After the students submitted their guesses, the average of all answers was calculated. The student who was closest to two thirds of the calculated average won the game. The first runs of this scenario were already done with the original *MobileQuiz* in 2013. At that time, the professor used a simple quiz with a single numeric slider input to collect the students'

guesses, exported the results into a prepared Excel sheet and calculated the winning number there. He then asked who guessed the right answer and handed a piece of candy to the winner.

In contrast to the *MobileQuiz*, the *MobileQuiz2* was able to execute the whole scenario within one application. Average and winning number can be calculated automatically as soon as the lecturer closes the quiz round, and the winning student gets a message displayed on his or her screen.

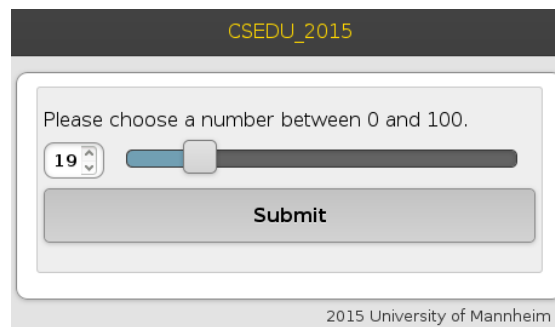


FIGURE 4.15: Screenshot of the students' view before submitting an answer.

4.3.7 Guess Answer

Another example from the area of game theory is a questionnaire with a ranged answer option. A professor in business administration wanted to do an experiment on the students' bias of overestimating their own knowledge. The students got ten questions about familiar facts of which details were unknown, e. g., how deep is the deepest point on earth or how much weight has a Boeing 747. The students had to guess a minimum and a maximum value, so that they had a ninety percent confidence that the right answer is located between their guesses. If they estimated right, nine out of ten questions should have been correct. The scenario appeared similar to the already known quizzes with open text answers, in spite of the fact that it had two answer elements per question. But the rules to determine the correctness of an answer were slightly more complex. This scenario used a specialized kind of question which could not be described with the mechanics of the basic *MobileQuiz*.

4.3.8 Survey

The *MobileQuiz2* can also be used to collect information via a survey or questionnaire scenario. This scenario uses a Likert scale question to ask opinions about a specific topic. The range of the scale can be changed individually by the lecturer. Every question can be adjusted with an individual range, e. g., a range from one to four or from zero to ten. In addition to Likert scale questions, the scenario has a multiple-choice question to pick several topics out of a list, and a question for open text answers as can be seen in Figure 4.16. An optional *description* field can be used to describe the frame of the current questions. This description then appears as a read-only label on the students' devices. The answers can be displayed as in the other scenarios but will in most cases be exported to an external data sheet for further research.

How important is it to you that the teaching scenarios are adaptable to your individual needs?

(1 = not important | 4 = very Important)

2

How important is it for you to have an overview of the current students' activities? (e.g. How many students already submitted)

(1 = not important | 4 = very Important)

3

Could you also think of using MQ2 for research purposes?

Yes, in some occasions.

FIGURE 4.16: Screenshot of the survey scenario with two Likert questions and a free text answer.

4.3.9 Further Variations

In addition to the already used scenarios which were mentioned above, the advanced model is able to describe more complex and innovative classroom interactions. With small improvements, the *MobileQuiz2* would become able to visualize the students' input of the "Message Board" scenario as a dynamic tag cloud instead of a plain list. The tag cloud could then be used as a feedback to students' opinion or constructive input. E. g., the scenario could contain various images which should be tagged by the students with topic related keywords. In another scenario, students might be enabled to write short medical reports, which are then made available for the whole course. The scenario would last for the whole semester in which the students were in practical work in hospitals. Thereby they would be informed about the medical projects of their fellow students. The collected reports could be discussed together at the end of the semester when students met with the lecturer. Without performance limitations, scenario would be able to contain so many rules that it would be able to simulate a complete marketplace. This scenario might be realized with one type of merchants or with two separate groups of buyers and seller. The scenario could be created with or without an almighty broker, dictating the prizes. In conclusion, the *MobileQuiz2* has the capability to perform teaching scenarios which with a didactic benefit beyond plain knowledge quizzes.

We now have presented the implementations of the lightweight *MobileQuiz* which is embedded as a plug-in in our university's learning management system ILIAS, and the customizable *MobileQuiz2* which enables lecturers to create scenarios fitting to their individual teaching styles. The various opportunities given by the *MobileQuiz2* were described, and some already used scenarios were presented. The next chapter describes

the evaluations of the tools. Both tools were evaluated by students and lecturers in small and large lectures for the last five years.

Chapter 5

Evaluation

The implementations presented in Chapter 4 were used to evaluate the two models with their different features. This chapter describes these evaluations and their results. The evaluations were focused on technical parameters, students' and lecturers' acceptance, and a basic view on the didactic usefulness. Log files and user statistics of the systems and their environments were analyzed for the technical evaluation, whereas the users' acceptance was evaluated with surveys and interviews by the students and lecturers.

5.1 Technical Evaluation

The implementations were designed for good usability and easy accessibility, but several technical challenges appeared during their usage. Among these challenges were the implementation of the QR code technology, the bandwidth requirements of the WiFi network, the distribution of mobile devices among the students, and the technical performance of the implementations. The investigation of these issues was mostly done without interviewing students or lecturers directly. The necessary parameters and information could be gathered through isolated experiments or could be extracted from the involved applications. The experiments and results described in this section have already been published (see [4–6, 11, 12, 14]).

5.1.1 Evaluation of the QR Code

Before using the QR code as the default method to access the quizzes, we had to analyze how well the scanning of the QR code would work under real classroom conditions. The appearance of the QR code displayed with a projector had to be considered in daylight conditions. In addition, the readability of the QR code had to be tested with different mobile devices and from different positions in the classroom. A QR code can be created in different sizes and with various error correction levels. The readability on four mobile devices were tested against several QR code sizes and error correction levels, as shown in Table 5.1. The tests were performed in a university classroom with 140 seats and typical daylight conditions. Every phone tried to scan the projected QR code from seven given distances between the first and last row of the classroom. The first row of seats had 1.5 meters distance to the screen, and the last row was located at 15 meters distance.

Problems occurred when scanning a large code from a side position of the first row, as the image had a strong perspective distortion from this point of view.

TABLE 5.1: Test devices and their camera resolution.

Device	HTC Hero	HTC Desire HD	iPhone 4	Samsung Galaxy S2
Resolution	5 megapixel	8 megapixel	5 megapixel	8 megapixel

When increasing the number of encoded characters or the level of error correction, the number of black and white squares of the QR code increases, and the size of the individual squares shrinks, which impairs the code's readability. The experiments showed that using a minimal level of error correction resulted in the best readability. This can be explained as the used method of error correction is designed to handle burst errors, which happen when the code loses a corner of the image or is partly covered by dirt. But these kinds of errors are unlikely to happen when the QR code is displayed on an indoor classroom projection screen.

Figure 5.1 shows the impact of the number of encoded characters on the code's readability with increasing reading distances. It can be recognized that the two devices with an 8 megapixel camera were able to read the 30 letter code from every position in the classroom, whereas the two 5 megapixel cameras failed at about twelve meters distance from the screen. Only one device was able to read all codes in every distance. Consequently, the implementations only encoded QR codes with a minimal level of error correction. In addition, to lower the number of encoded characters, a URL shortening service was used to minimize the number of characters of the original URL.

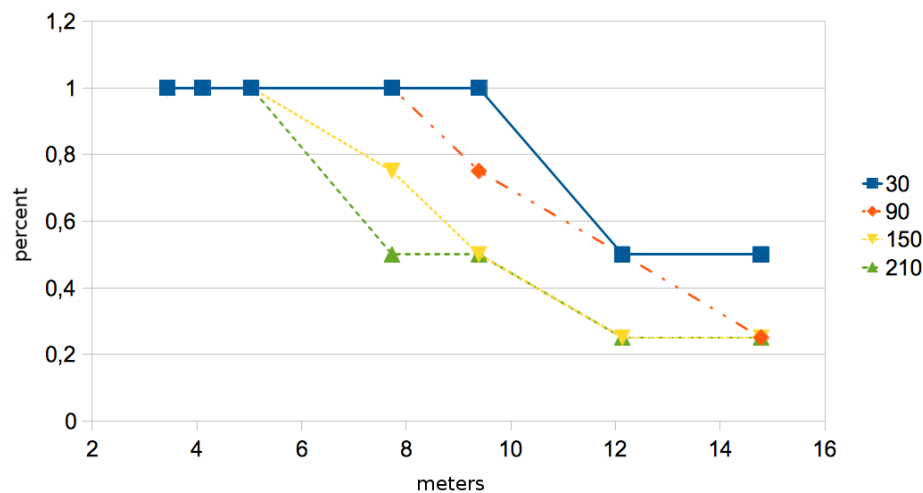


FIGURE 5.1: Percentage of smartphones which were able to read a QR code of 30, 90, 150 and 210 characters at different distances.

5.1.2 Workload on the Network

Besides the readability of the QR code, difficulties with the WiFi network were identified. As one of the specifications was to avoid setting up a dedicated network, all devices had to use the existing university WiFi network or the data network of a mobile network

provider. Unfortunately, the WiFi network at the university was not designed to serve more than a hundred parallel data connections within one room at that time. Therefore, it was one of the first topics of interest to survey the network load and network delay.

In a sample experiment, the network load caused by a single student using the *MobileQuiz* was measured. Afterwards, the actual load in real classroom conditions was measured during a lecture with 139 students. After implementing the *MobileQuiz2* and recognizing new performance issues, the measurements of the WiFi network traffic were repeated. We report the results below.

Workload in a Sample Study with the *MobileQuiz*

In this sample experiment, the network load caused by one student was measured in an isolated environment. The experiment was performed with the *MobileQuiz* and a quiz containing five questions with three possible answers each. The answers contained no pictures or other multimedia content. Figure 5.2 shows the network load for this quiz participant. The data download to the device caused about 240 kbits, which is represented by the high bar at the sixth second. The upload for submitting the answers caused about 100 kbits of traffic at second 15. Altogether, 340 kbits of network load arose for one participant. When inspecting the data, it could be recognized that the largest part of the downloaded data consisted of graphic symbols, which were used for the layout and the JQuery libraries.

After this measurement for one participant, the strain of the WiFi network was measured with the university WiFi network access points. While the quiz was used during a lecture where seven devices participated, the load on all nearby WiFi access points was measured for the whole lecture time. But the additional network traffic of these seven devices could not be recognized in the overall daily WiFi network traffic, it was absorbed by general traffic noise. The WiFi access points could handle up to 100 mbit/s, so even for more than 100 students the pure data load should not have been a limitation. Unfortunately, independent of the network load, the WiFi network delay increases with an increasing number of clients (see also Xiao [209]). Additional packets are needed to regulate the network traffic and to avoid packet collisions. This causes a noticeable packet overhead and increases the overall delay. But considering the small number of students in this sample test scenario, an impact on network delay could not be noticed.

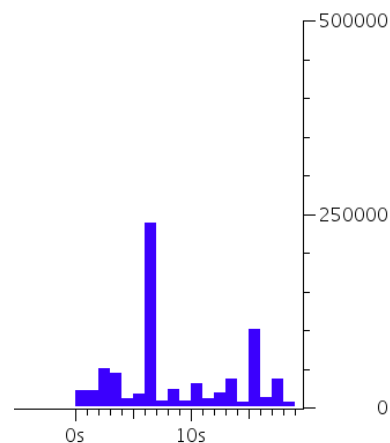


FIGURE 5.2: Network load for one quiz round in bits/s when using the *MobileQuiz*.

Workload in Classroom Study with the *MobileQuiz*

After the sample study described above the network load was tested in a larger classroom environment. The network load was measured a second time, with the university's WiFi access points during a quiz round. But this time, a lecture was selected which was attended by 139 students. Figure 5.3 shows the amount of network data during the entire lecture time. Upload (blue) and download (red) were measured in megabits per second and arithmetically averaged over a frame of four seconds. The quiz took place from 10:36 to 10:41 and consisted of only one numeric question. 95 students participated, and 85 of them used the university's WiFi network. The other ten students used the Internet access of their mobile phone network provider. At the time when the quiz started, the graph shows a peak up to 8 Mbit/s and then decreases to about 3 Mbit/s. When comparing this amplitude to the average network load during the whole lecture, the quiz can clearly be recognized. But regarding the maximum possible network load of 100 Mbit/s (IEEE 802.11n with MIMO) for our university's WiFi, the maximum capacity of the WiFi network was not utilized. But as mentioned above, the WiFi network delay increases with an increasing number of clients, because additional packets are needed to regulate the network traffic and to avoid packet collisions. Even with about 85 students participating via WiFi, a significant impact on the network delay was not recognized. However, only quiz rounds with plain text questions were examined so far. When adding multimedia content, the network load would rise considerably.

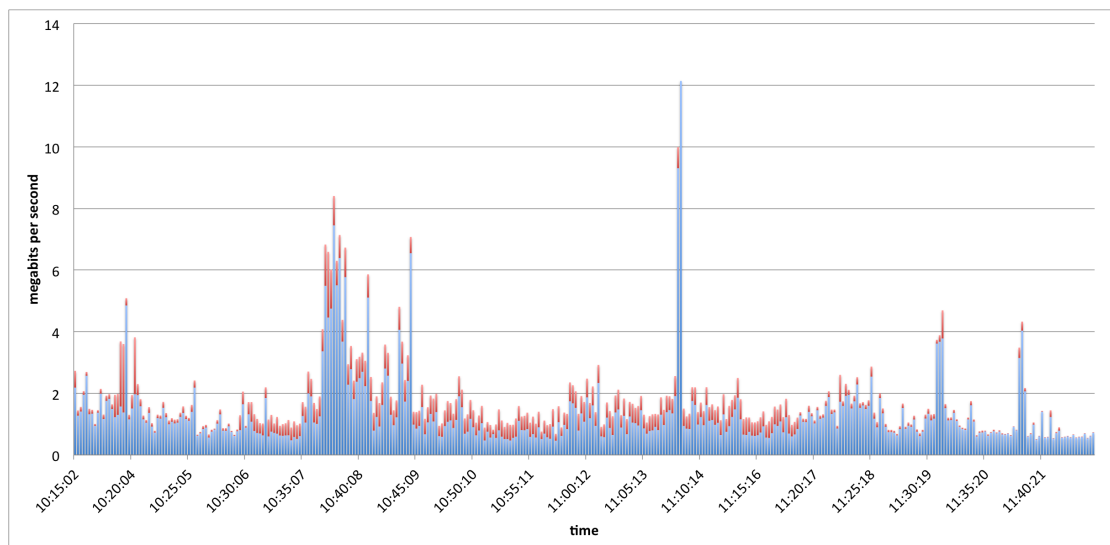


FIGURE 5.3: Network load during a *MobileQuiz* usage in a lecture with a quiz at 10:36h.

The experiences show that the *MobileQuiz* application was feasible in practice and met its requirements. The measurements on the network load showed that the university's WiFi infrastructure was sufficient; lectures with hundreds of students could be handled. The reason why ten students used the Internet connection of their mobile phone provider was not further investigated. Whether it was their default access option or if they had trouble connecting the university's WiFi network could not be told.

Network Load when Using the *MobileQuiz2*

The analyses with the *MobileQuiz* showed, that the network was no bottleneck. This observation was confirmed by several semesters in which the *MobileQuiz* was used during actual lectures. Even in classrooms with more than a hundred of students, no performance issues emerged. But this changed when lecturers started to use the new *MobileQuiz2*. Depending on the scenario, huge performance delays could be observed, especially in lectures with a high number of participants. To identify the new bottlenecks, the network traffic was observed again. At first, it was measured directly on the Apache web server.

Figure 5.4 shows the number of server connections in a large exercise in the Fall semester 2015. It was a scenario of *Question Phases* with three phases; about 80 students participated. The three phases can easily be identified in the Figure by the peaks in the number of connections. The server's configuration for the maximum number of connections was set to 150. Unfortunately two separate connections were established for every participant although only one was actively used. During the third phase, the server became so busy that it wasn't even able to log the connections any more, and it took about ten minutes to recover. Because the students complained about the huge delays, the lecturer canceled the quiz round at that point.

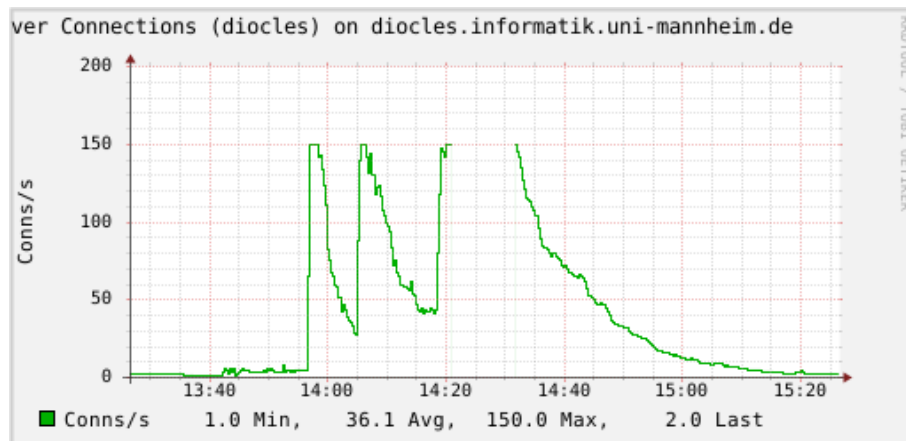


FIGURE 5.4: Server connections during a complex *MobileQuiz2* scenario with a gap at 14:20h due to CPU overload.

When looking at the actual network traffic in Figure 5.5 the view slightly differs from the information above. In the first two phases, the network traffic rose up to 800 kilobytes/s, which describes the students' activity as expected. But in the third phase, when the server got really busy, the actual network traffic at the server remained under 400 kilobytes/s and even decreased shortly before it got too heavy at 14:20. When considering this observation, the network load is less likely the most restraining bottleneck.

In search of the real bottleneck, the application logged delay times for the whole execution process, which was described in detail in Section 3.4.5. This process was further divided into four steps:

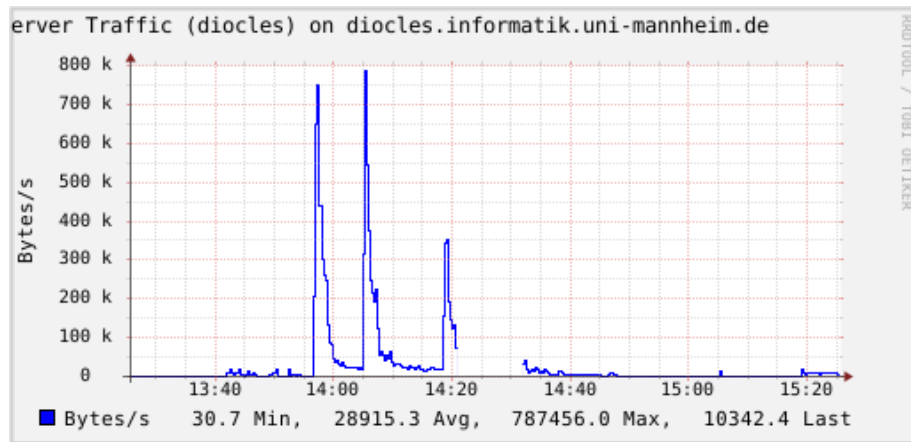


FIGURE 5.5: Network traffic on the server during a complex *MobileQuiz2* scenario with a gap at 14:20h due to CPU overload.

- Time in the network.
- Time for processing the data.
- Time to gather the relevant data attributes for the student's display.
- Time to render the page with the new information on the student's phone.

The times were measured for different kinds of input actions, as displayed in Figure 5.6. The Figure shows the mean values of the measured times throughout several *MobileQuiz2* quiz rounds. It could be observed that only a small amount of delay was caused by the network. Most of the delay was caused by data processing on the web server. The delays for gathering the information attributes or rendering the layout on the mobile devices were so small that they could almost be ignored. This suggests that the bottleneck has to be identified within the server's data processing.

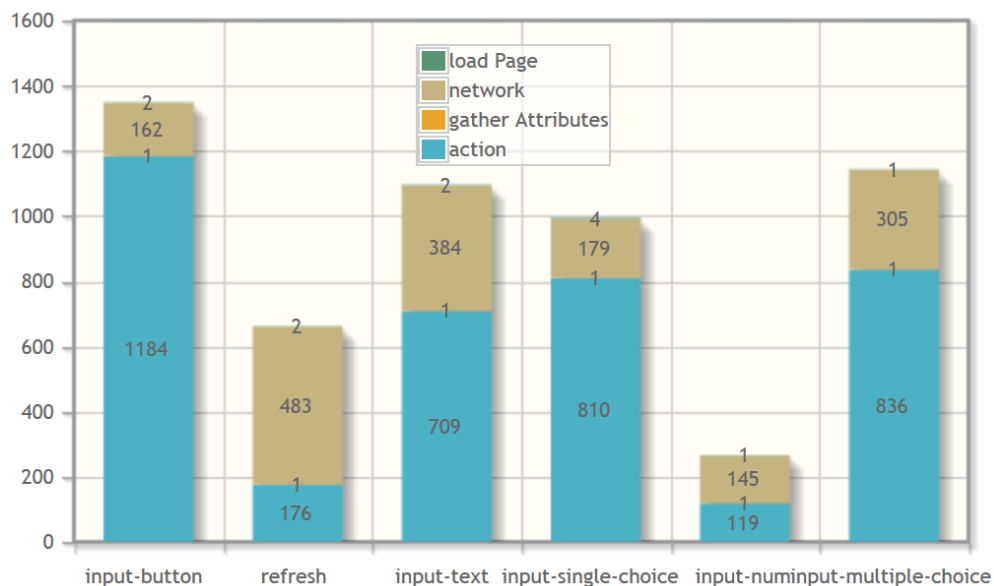


FIGURE 5.6: Broken-up time delay of different *MobileQuiz2* inputs.

5.1.3 Server Performance in a Classroom Study with the *MobileQuiz2*

The results obtained so far imply that the main reason for the time delay when using the *MobileQuiz2* is the servers performance. Therefore, the CPU usage, memory utilization and system load were measured, too. Figure 5.7 shows the CPU usage of one of the four cores of the used server. The CPU-workload was similar on all four CPU cores, therefore only one core is displayed here. It shows three peaks of CPU usage at 14:00, 14:10 and 14:20, similar to the peaks of network load above. The third peak is much broader than the other two. It is as broad as the time in which Figure 5.4 and Figure 5.5 show their measuring gap.

In addition to the CPU usage, the memory utilization was measured, as shown in Figure 5.8. The picture is quite similar to the already presented Figures. The three phases are easily recognizable as peaks, whereas the third peak is broader than the two before. But considering the full amount of available memory, the memory was not strained very much.

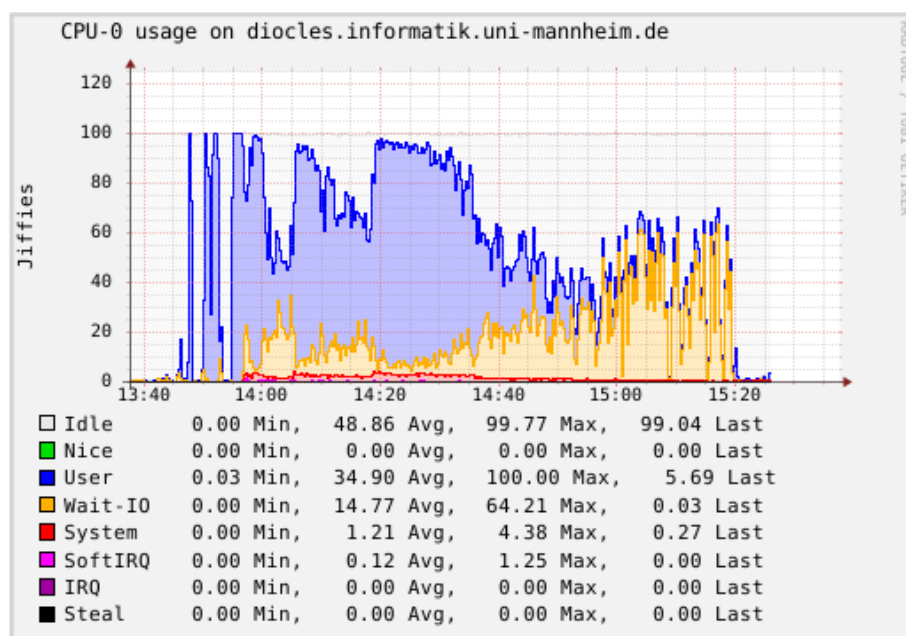
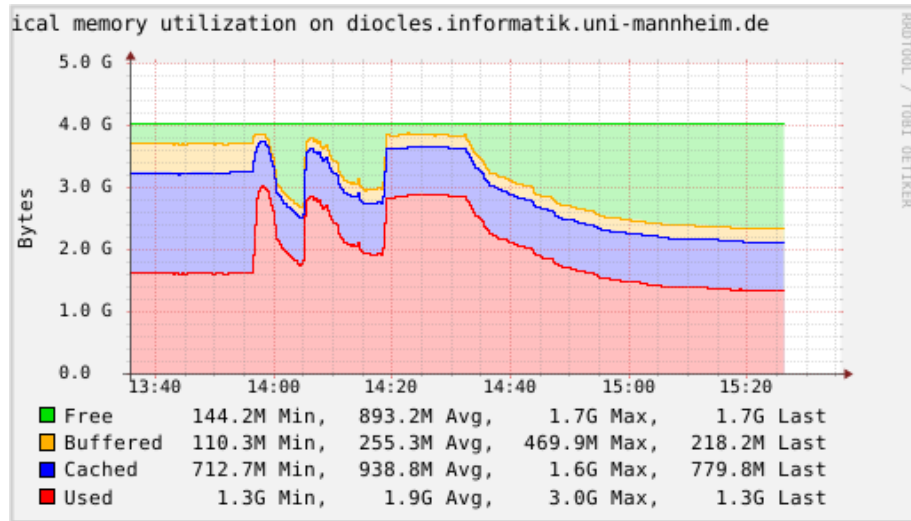
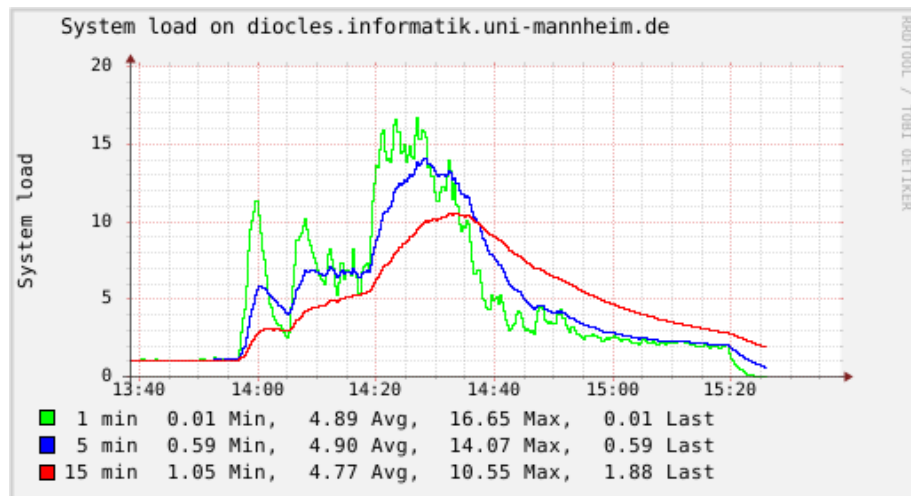


FIGURE 5.7: CPU usage on the server during a complex *MobileQuiz2* scenario.

Figure 5.9 shows the system load on the server. Having four cores, the server could handle a workload up to a total load of four. A system load above four means that the server is not able to compute all processes at once, and some processes have to wait. Figure 5.9 shows that the average load raised up to 17. Even load averaged with a 15 minutes frame reached a value of ten. That means that on average, six processes were idle and had to wait until one of the four currently computed processes finished. Therefore, the bottlenecks are most probably the application's algorithms.

5.1.4 Server Performance when Using Various Scenarios

After identifying the algorithms as the crucial bottlenecks, it had to be figured out which scenarios caused the problems. Therefore, the *MobileQuiz2* was extended with a test module to simulate several scenarios under classroom conditions. The simulator

FIGURE 5.8: Memory usage on the server during a complex *MobileQuiz2* scenario.FIGURE 5.9: Load on the server during a complex *MobileQuiz2* scenario.

could create a given number of participants with distinct *GameObjects* and *GameAttributes* and was able to simulate the performance of rules on these elements. Three scenarios were chosen to create a better understanding of the possible performance issues. Table 5.2 shows these three scenarios with their number of rules and the number of *GameObjects* and *GameAttributes* when used in a lecture with 100 participants. A simple, medium, and quite complex scenario were chosen from Section 4.3 to test the processing of the rules.

TABLE 5.2: Test scenarios with number of rules, *GameObjects* and *GameAttributes* with 100 Players.

Scenario	Rules	GameObjects	GameAttributes
MessageBoard	2	102	302
MobileQuiz	1	1400	3200
QuestionPhases	5	301	2101

As a simple scenario, the *Message Board* was chosen. It only has two rules and the

smallest number of *GameObjects* and *GameAttributes*. The medium scenario was a *MobileQuiz* with three questions and three answers each. It only had one rule but the highest number of objects. The most complex scenario was the *QuestionPhases* with three questions and three phases.

Figure 5.10 shows the time in milliseconds, which the server needed to create the *GameObjects* and *GameAttributes* for a rising number of players. The simulation was done for one up to one hundred players with steps of ten. Even for the scenario with the smallest amount of objects, the server took about 25 seconds (25000 milliseconds) to create all 102 *GameObjects* and 302 *GameAttributes* for the 100 players. To create the elements for the medium complex *MobileQuiz* scenario, it even took more than four minutes. The creation of the elements of the complex scenario took about two and a half minute (137400 milliseconds). In this test, the complex scenario took less time than the medium scenario, as it needed a smaller amount of *GameObjects* and *GameAttributes*. As can be anticipated from the Figure 5.10, the algorithm has a complexity of $\mathcal{O}(n)$, but nonetheless, the amount of time to only create all necessary elements for a scenario could overstress the server, as many participants usually enter the quiz at once.

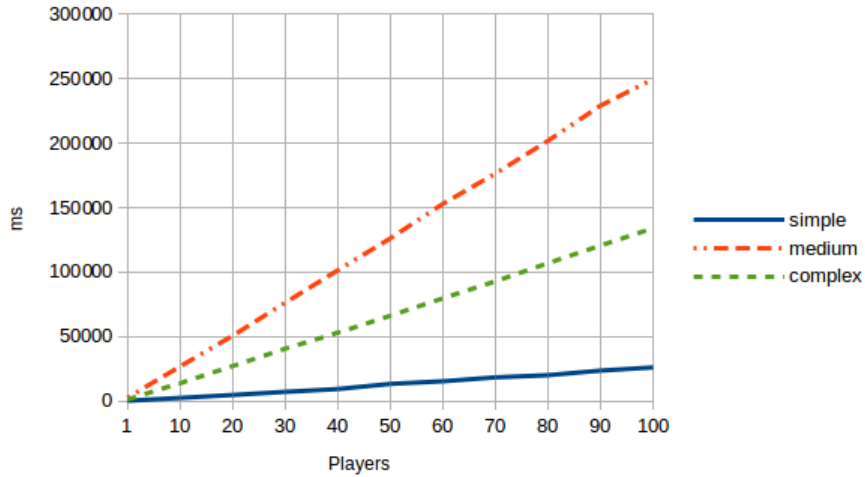


FIGURE 5.10: Times for object creation in a simple, medium and complex scenario.

Regarding the processing time for the rules, another simulation was done. We measured the time it took to perform a rule on the elements created in the experiment above. Figure 5.11 shows the times for the medium scenario. Similar to the creation of the game elements, rule processing seems to have a complexity of $\mathcal{O}(n)$ and therefore rises linearly. The other scenarios showed a similar gradient in their curve, but had a different offset at the start. This can be explained by the different types of the rules checks. When reaching 100 players, one rule iteration of the medium *MobileQuiz* scenario took about 217 milliseconds.

Considering, that the rules are performed every time an input is triggered by any participant, the frequency of triggers also increases with $\mathcal{O}(n)$. Therefore, the overall complexity is $\mathcal{O}(n^2)$, whereas n is the number of participating players.

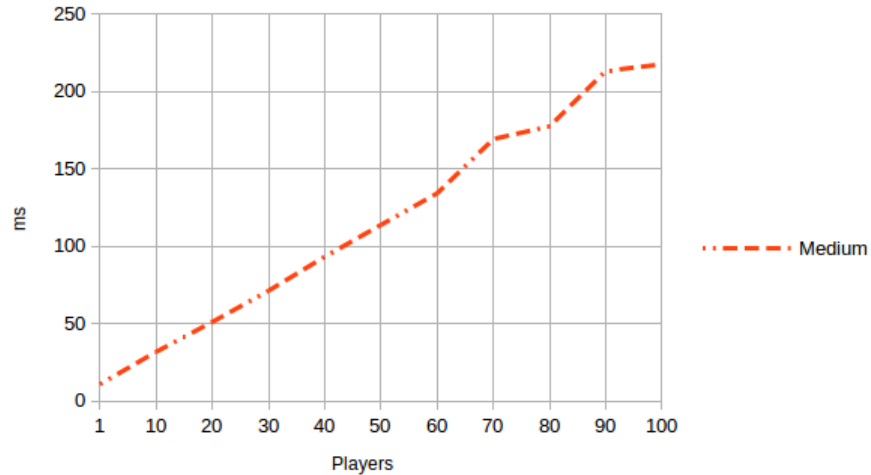


FIGURE 5.11: Times for rule performance of a medium scenario.

5.1.5 Distribution of Students' Smartphones

Besides specially designed evaluations, the *MobileQuiz2* logs anonymous data. Among this data is the operation system, which is installed on the students' devices. This data offers only a minor value for the evaluation of teaching scenarios, but it can be useful for application development and maintenance.

Figure 5.12 shows the aggregated number of operating systems of all participants in the Fall semester 2015. 55% of the students used an Android device whereas 25% used an Apple product. Among these are about 4% using an iPad. 21% of the accesses were done via a Windows system. This includes Windows phones as well as laptops and PCs with a Windows operating system.

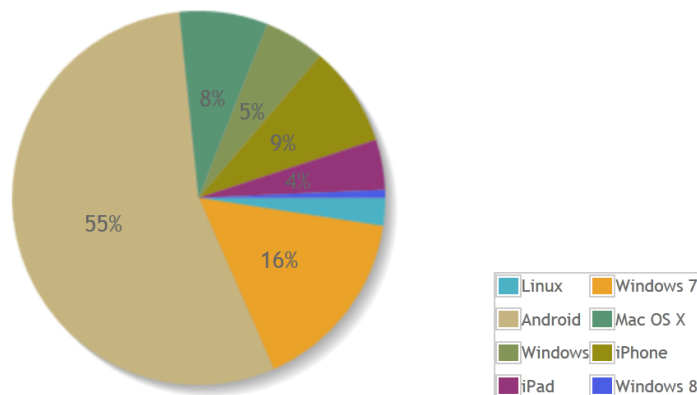


FIGURE 5.12: Distribution of operating systems of students in the Fall semester 2015.

5.2 Evaluation of Acceptance

In addition to the technical aspects of the implementations and their bottlenecks, the users' acceptance was a main topic of interest. The focus was the students' acceptance of the implementations and the lecturers' adaption of the technologies.

Several evaluations with different patterns were used to question students and lecturers. The students' opinions were mainly collected with quantitative surveys in the classroom, whereas lecturers were mostly questioned with open questions to get a qualitative feedback on the concepts and implementations. The content of the questions ranged from technical performance issues to didactic challenges. The evaluations were not conceived to answer didactic hypotheses.

5.2.1 Acceptance by Students

Pre-Study of the QR code's Acceptance

In addition to the technical parameters of the QR code evaluation in Section 5.1.1, the *MobileQuiz* was evaluated in an actual lecture in the Spring semester of 2012 [6]. The students were asked one week before the evaluation to bring a web-enabled mobile device into the lecture. They were informed about the new system just before they participated in two quiz rounds. Unfortunately, because of the upcoming Easter break, this lecture was poorly attended (by only nine students). Only seven students participated in at least one quiz round. Those who did not bring a smartphone with a QR code reader used their regular phones or laptops and entered the URL of the quiz round manually. The two remaining students stated that they could not participate in the quiz because they left their devices at home or their current battery level was too low. The results supported the statements of the lecturers who said that almost all students own a web-enabled device which is suitable for the quizzes. Except for one student who used the data connection of his mobile network provider, all participants connected via the university's WiFi. Six students answered that the usage was fast and uncomplicated. Only one, who manually typed in the link by using an old Sony Ericson phone, stated the usage as complicated and slow. Nobody spoke against the quiz, and four students wanted to see the quizzes more regularly in their lectures. The positive effects on the students' motivation and learning success, which were already analyzed by previous research (e.g., Scheele et al. [177]), could be confirmed. Although the authors used different types of devices and technologies, the pedagogical impact seems quite the same.

Evaluation of the *MobileQuiz* in a Large Lecture Scale

After a proof of concept had been conducted in a lecture with a small number of students, the tool was evaluated in two larger lectures with 253 students altogether [2, 4]. The lecture was held in the Fall semester 2012 on the topic "Business Economics". The course was held in two parallel lectures with about 130 students each, and the students were asked to participate in a quiz every second week during this semester. The evaluation was not announced beforehand and the students had to fill in a paper questionnaire at the end of the lecture. 201 out of 253 students participated in the quiz, 52 did not (the questionnaire and original data can be found in Appendix B). Table 5.3 shows the stated

reasons for not participating in the quiz. With 62.7%, most of the 52 students did not participate because they did not have their devices at hand. About 16% said that they wanted to participate but were not able to do so, mostly because of the unstable WiFi network connection. About 4% said the participation was too troublesome. Only 9.8% said that they do not have a suitable, web-enabled mobile device.

TABLE 5.3: Students' reasons for their lack of participation.

Reason	%
did not work	15.7
too troublesome to participate	3.9
brought no device with me	62.7
own no device	9.8
other	7.8

The 201 participating students mostly used their smartphones (88.4%). Only 6.6% used other devices like iPads. 4% used their laptops, and only one percent used an ordinary mobile phone (see Table 5.4).

TABLE 5.4: Distribution of the students' devices.

Device	%
smartphone	88.4
laptop	4.0
ordinary phone	1.0
other	6.6

One of the survey's goals was to figure out how fast and easy the quiz access was perceived by the students. Another was to find out if they thought the *MobileQuiz* application increased their motivation or understanding of the lecture content. Almost sixty percent of the students said the access was "fast" or "really fast", as shown in Figure 5.13. 27.8% said it was "ok", and only 13% were unsatisfied with the access speed.

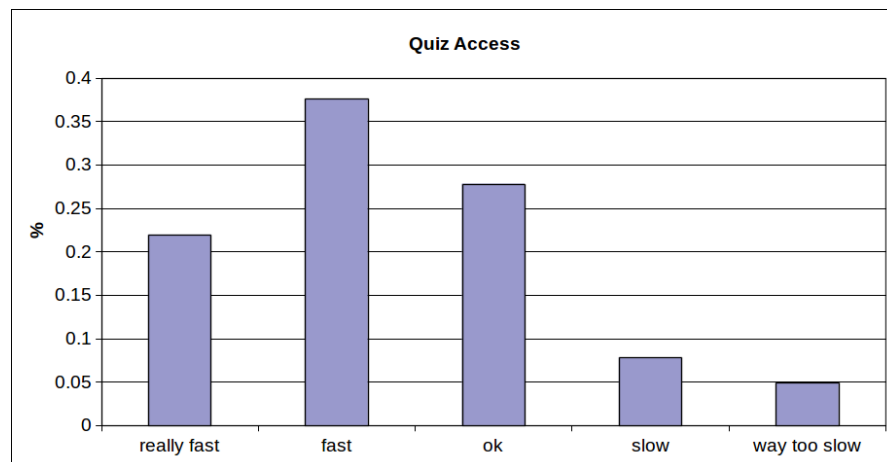


FIGURE 5.13: Students' evaluation of the speed of quiz access.

When asked about the technical difficulties, more than fifty percent said that the *MobileQuiz* application is really easy to handle (Figure 5.14). Other students answered that the handling was "easy" (31%) or "ok" (12%). Only three percent perceived the participation in the quiz as difficult.

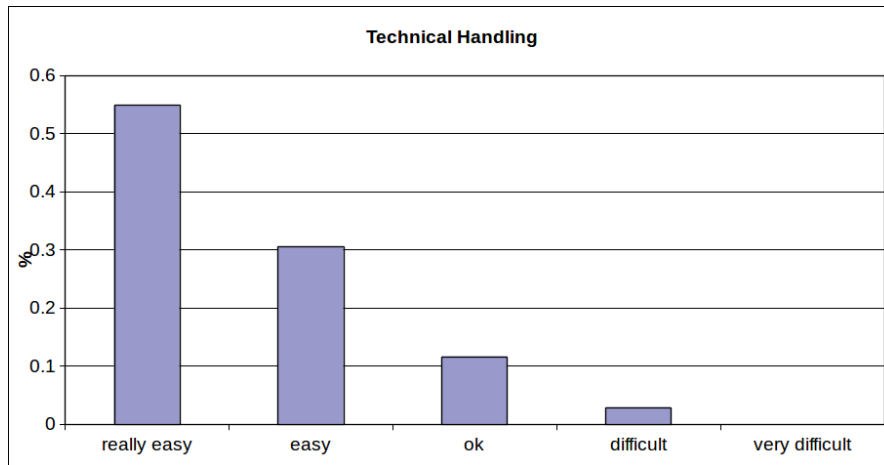
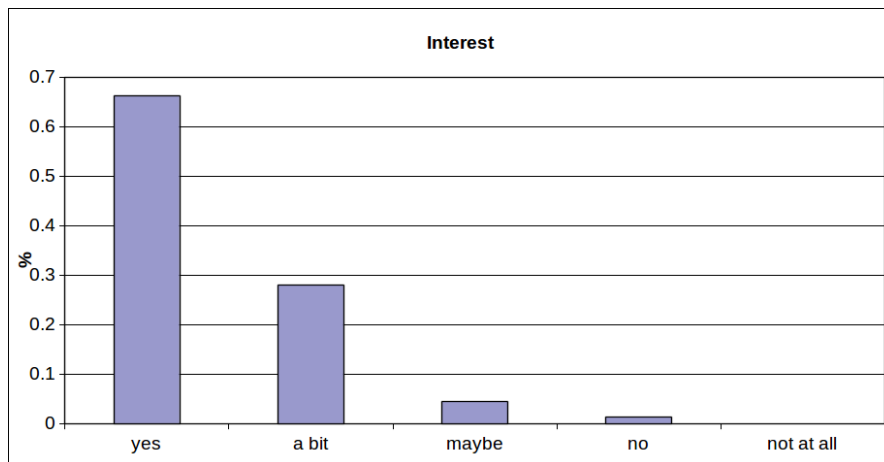


FIGURE 5.14: Students' evaluation of the ease of quiz handling.

Furthermore, the students were asked how they estimated the impact on motivation, content understanding, and increasing the interest in the lecture. Figure 5.15 shows that fifty percent of the students said that the quiz actually increased their motivation. 19% did not know if there was an impact on their motivation, and 27% said that their motivation was not increased. When asked if the quiz increased the interest in the lecture, 66% answered with “yes” and 28% with “a bit” (see Figure 5.15). Only 1% said that the quiz does not increase their interest in the lecture.

FIGURE 5.15: Students' answers if the *MobileQuiz* increased the interest in the lecture content.

About 76% of the students thought that using the quiz in lectures helped them to better understand the lecture's content. Only 7% said it had no impact on the understanding (Figure 5.16). Finally, the students were asked if they wanted to see such quizzes more frequently. 68% answered with “yes”, 26% said “maybe” and only 6% did not want the quiz rounds more often.

Considering the question “did the quiz make the lecture more interesting for you?”, 94% of the students agreed (66% “yes”, 28% “a bit”). 77% of the students agreed on the question “did the quiz help you to understand the lecture content?” (33% “yes”, 44% “a bit”). In contrast, the answers to the question about the student's motivation had

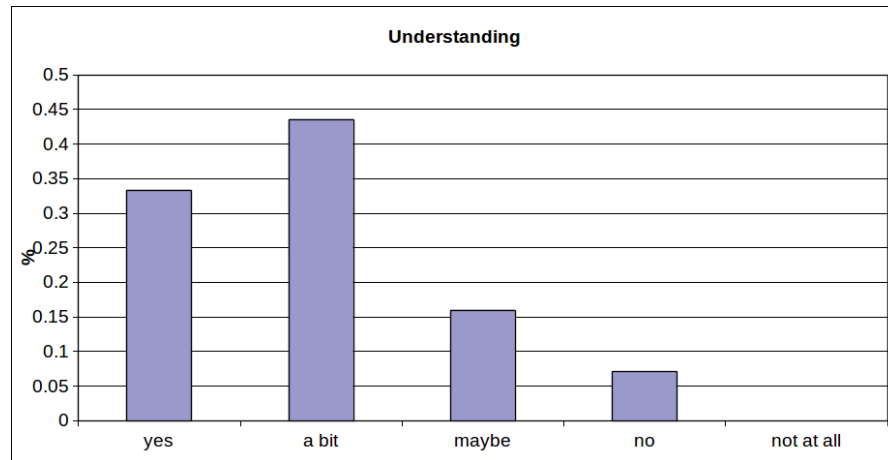


FIGURE 5.16: Students' answers regarding the understanding of the lecture content.

a high variance, as shown in Figure 5.17. Only 50% of the students said that the quiz actually increased their motivation.

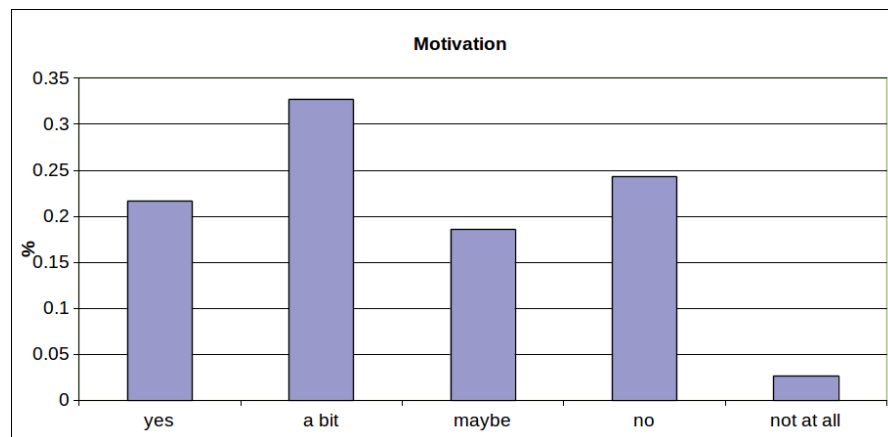


FIGURE 5.17: Answers to the question if the quiz increased the students' motivation.

One possible explanation is the little weight of a small tool like the *MobileQuiz* in a complex course design. Another possible reason is that learning motivation stems from many different aspects, e.g., interest in the topic, teaching skills of the lecturer, the students' mood, and the classroom atmosphere. Nonetheless, the strong agreement on the question whether the quiz should be used more often during lectures implies the high interest and acceptance of the tool, as shown in Figure 5.18. Almost 70% of the students wanted to see the quizzes more regularly and less than 10% did not want to use such quizzes again.

Evaluation of the *MobileQuiz* with a Control Group

The evaluation results of the *MobileQuiz*'s acceptance showed that about 80% of the students participated in the quiz rounds. Only two percent stated that they did not own an appropriate mobile device, and most students said that the technical handling was quite easy and fast. These results could be confirmed by an evaluation during a course of German literature (see Klinger et al. [2]) in the Fall semester 2012 (the original data

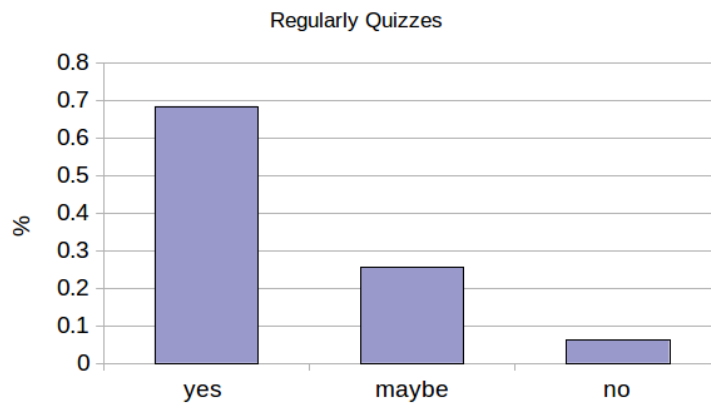


FIGURE 5.18: Students' answers to the question if quizzes should be used more often.

can be found in Appendix B in Figure B.6 and Figure B.7). The lecture was attended by 39 students, who were separated into two parallel groups. The first group with 20 students used the *MobileQuiz* at least one time every week, whereas the other group of 19 students did not use the *MobileQuiz* at all. But the lecturer asked the same questions in a classic oral way for the second group. This control group setting was chosen to detect biases in the lecturer's estimation of the students' knowledge and the usefulness of the *MobileQuiz*. Additionally, this setting also allowed to compare the students' exam grades with and without the quiz. On the question if the *MobileQuiz* increased their interest, the students did not show a clear opinion. Figure 5.19 shows that seven students stated that the course became more interesting when using the *MobileQuiz*, and four "agreed" to that statement. The other seven students only partly agreed or did not agree. Indeed, one student absolutely disagreed.

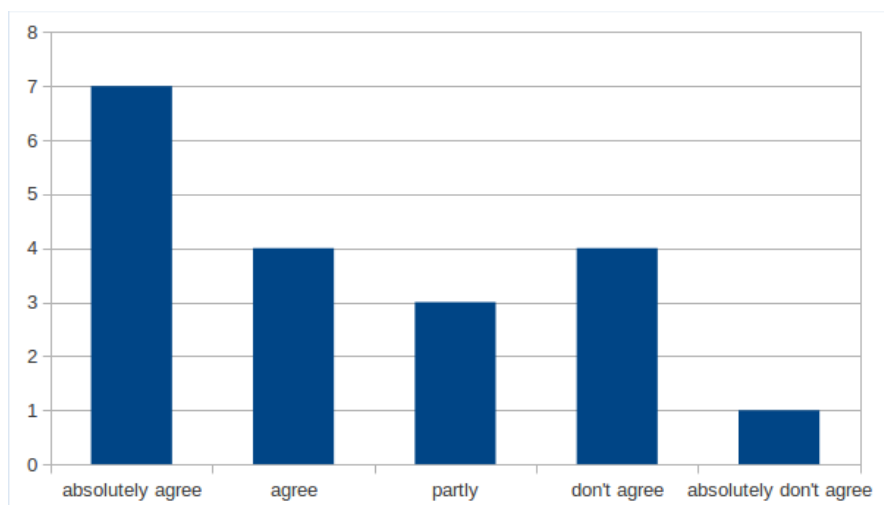


FIGURE 5.19: Students' answers if the course became more interesting due to the use of the *MobileQuiz*.

The surveyed students did not tend to say that their motivation to learn the course's content was increased by the quiz usage. Figure 5.20 shows that only one student absolutely agreed with that statement. Four students still agreed with it and six students stayed neutral. But eight students disagreed with it.

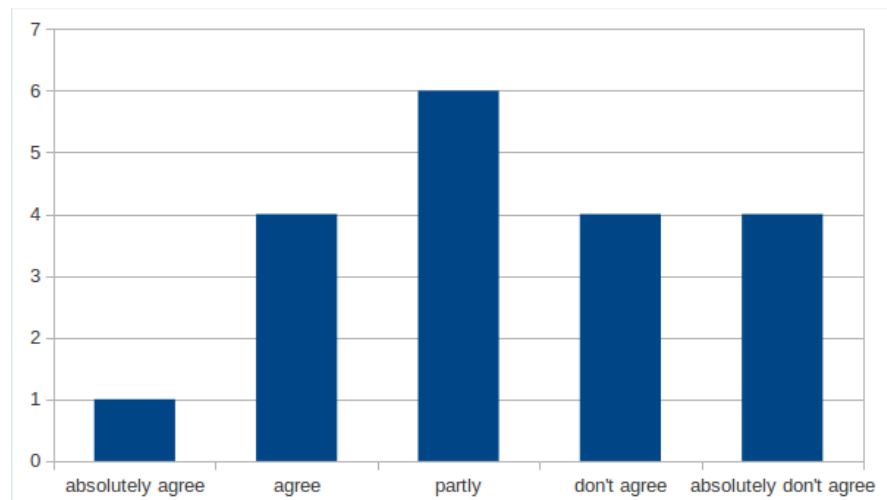


FIGURE 5.20: Students' answers if the *MobileQuiz* increases the motivation to learn the content.

On the question, if the quiz usage helps to understand the topic, the answers were heterogeneous again. Figure 5.21 shows that slightly more students disagreed with the statement. Six students answered neutral, whereas five students tended to agree and six students tended to disagree with it. When regarding the mean value of the grades of the course's exams, the group which used the *MobileQuiz* was slightly better than the control group. However, the results were not significant, as the number of participants was too low for the small differences in grade values. When summarizing this evaluation, it can be stated that the students welcomed the usage of the *MobileQuiz* as it enriched the course, but did not recognize a major didactic benefit.

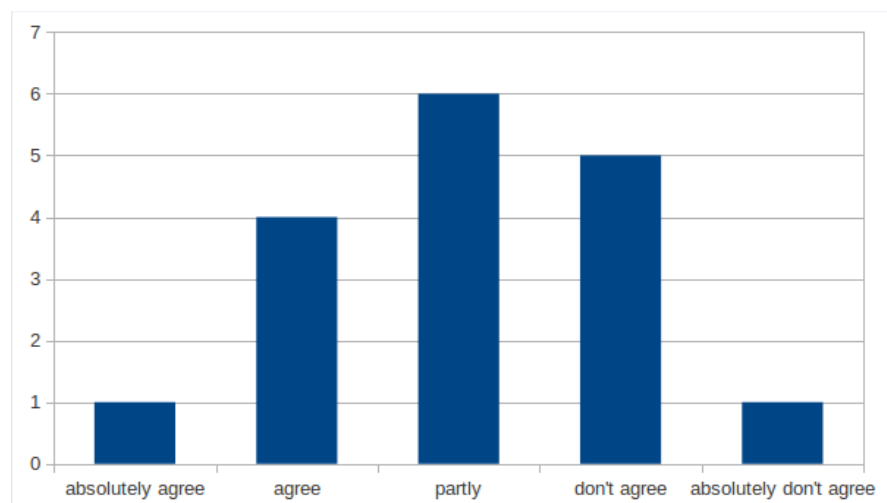


FIGURE 5.21: Student's answers if the *MobileQuiz* helped them to understand the content.

Evaluation of the HomeQuiz Feature

The *HomeQuiz* feature enables the students to use the *MobileQuiz* outside of the classroom. They are able to repeat the quizzes as often as they wish and get a direct feedback

on their results. Two courses were selected for a *HomeQuiz* test usage in the Spring semester 2013. After the test, a survey was conducted with the participating students and the lecturers were interviewed (originally published in Schön et al. [8]; the complete surveys can be found in Appendix C).

Course 1 (ACN): The lecture and tutorial of “Advanced Computer Networks” (ACN) was attended by approximately 70 students on the master level. ACN is one of the computer science fundamentals, and students typically participate in this course during the second semester of their master program. The students had already used the *MobileQuiz* during the past lectures, and some students asked for an offline version of the quizzes to repeat some of the topics. The quiz for the *HomeQuiz* test consisted of four questions and was added to one of the weekly exercise sheets. The students performed the quiz individually in the time between two lectures and were asked to fill out an evaluation form about the *HomeQuiz* feature at the beginning of the next exercise.

Course 2 (Literature): “Introduction to German Literature – Part 2” was attended by 18 students on the bachelor level. The function of the *HomeQuiz* was explained briefly, and a sheet of paper with the QR code tag and the URL to the quiz were handed out to the students (as it was a test usage of the functionality shortly after development, it was not possible to embed the QR code and the URL into the regular paper script). The students were asked to use the quiz function until the next classroom session. In the next session, a paper questionnaire was handed out and answered by the students.

In course 1, 46 questionnaires were collected (see Table 5.5), whereof almost ninety percent used the quiz and could be described as engaged as 77.3% tried to answer all questions. 76.7% of the participants could conduct the quiz without technical problems, 18.6% of the students had a few problems, and in 4.7% of the cases the quiz did not work. When given a list of possible problems, the results were as follows: “barcode reader did not work” (4.3%), “QR code unreadable” (2.2%), “access very slow” (4.3%), and 15.2% of the students voted for “other”, not precisely characterized problems.

TABLE 5.5: Participation in the *HomeQuiz*.

	Did you use the quiz?	Did you try to answer all questions?
Course 1 (ACN, $n = 46$)	89.1%	77.3%
Course 2 (Literature, $n = 14$)	50%	70%

Figure 5.22 shows that about 75% of the students “agreed” or “strongly agreed” that their motivation to prepare the course content was increased, and about 80% of the students answered, that the quiz increased their interest in preparing the course content (see Figure 5.23). Figure 5.24 shows that the majority of students think that the quiz helped them to recognize deficits in understanding the course content.

At least 50% were motivated to ask more questions in the following lecture (see Figure 5.25). When asked if they want these questions more often in the future, 51% “strongly agreed”, 40% at least “agreed” and less than 10% “disagreed” or “strongly disagreed”. The questionnaire ended with an open question concerning further comments or improvement proposals. The students listed that a quiz should cover only one

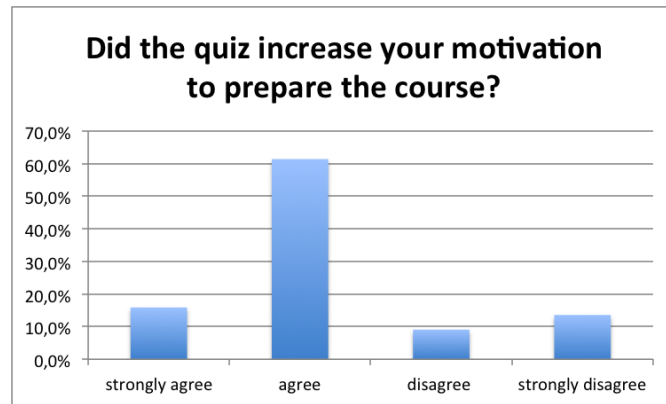


FIGURE 5.22: Evaluation of the question if the *HomeQuiz* did increase the motivation to prepare the course content.

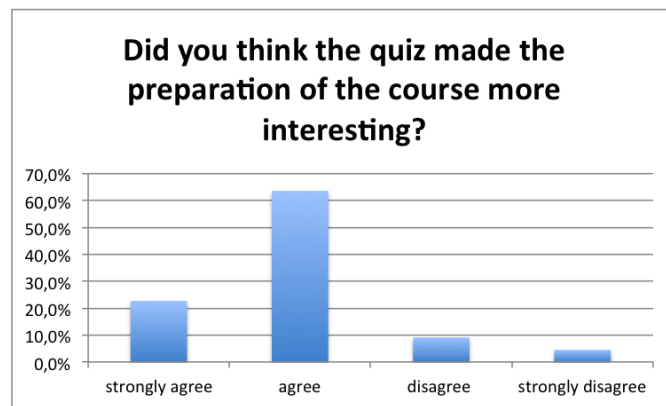


FIGURE 5.23: Evaluation of the question if the *HomeQuiz* made the course preparation more interesting.

particular topic to be less confusing, and that they wanted to see the overall answers to compare themselves to the other students.

The results of the survey in course 2 were less positive. 14 students attended the session and completed the questionnaire. Only seven used the *HomeQuiz* and gave their opinion. Six of seven students used the quiz without any technical problems. One student indicated minor problems because of a malfunctioning barcode reader. All seven students disagreed to the question if the quiz increased their motivation to prepare (or respectively rework) the course.. On the question if the quiz made the preparation/rework more interesting, two students “agreed”, whereas one “disagreed” and four “strongly disagreed”. Detected understanding deficits did not motivate students to ask the lecturer in the course as all seven disagreed. At least, three students confirmed that the quiz helped them to detect deficits in understanding, whereas one “disagreed” and three “strongly disagreed”. On the question, whether students thought the quiz should generally be used more often for preparation and rework, one student “strongly agreed”, one “agreed”, four “disagreed” and one “strongly disagreed”. Asked, if more questions in one single quiz round are preferred, two students “strongly agreed”, one “agreed”, two “disagreed” and two “strongly disagreed”. In summary, the results of course 2 are - besides the question of representativeness - rather negative concerning the *HomeQuiz*.

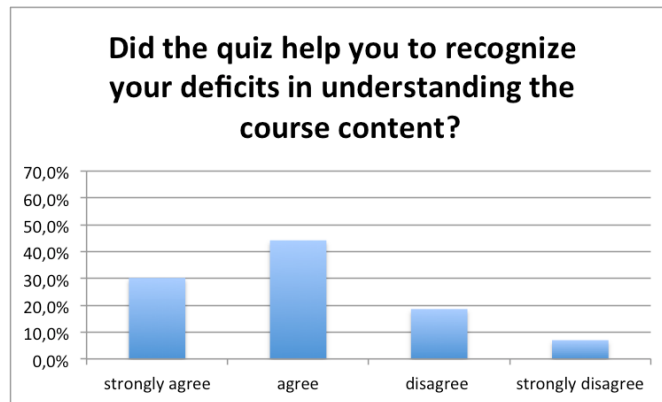


FIGURE 5.24: Evaluation of the question if the *HomeQuiz* helped to recognize the deficits in understanding the course content.

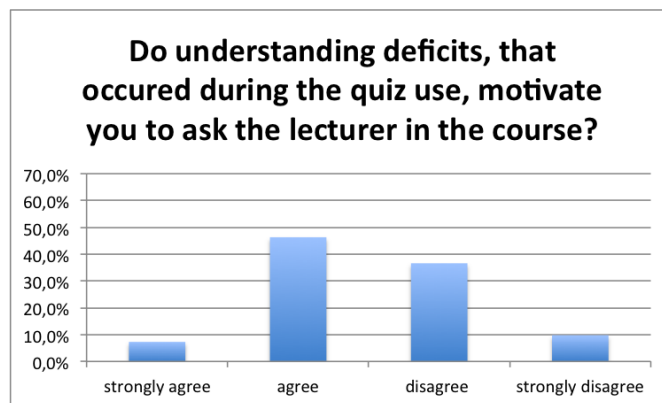


FIGURE 5.25: Evaluation of the question if the usage of the *HomeQuiz* increased the students' willingness to ask questions in the next lecture.

The results show that the students in course 1 generally accept the *HomeQuiz* functionality and mainly like to use it for verifying their state of knowledge. Above 75% of the students in course 1 perceived an increased motivation and interest in course preparation. Above 70% of them said that the quiz helped them to recognize their knowledge deficits and at least 50% were motivated to ask the lecturer questions about their deficits in the next lecture. This relatively low percentage could indicate that the students can identify their problems in understanding by themselves when given feedback about the correct and incorrect answers by the system; they do not need additional support by the lecturer. Moreover, some of them could still be afraid to ask the lecturer and thereby embarrass themselves. The result could also hint on alternative learning strategies: students presumably ask their fellow students or try to find the solution to their problem within books or the Internet. Still, the interactivity in class grows significantly, if 50% of the participants are motivated to ask further questions. Finally above 90% of the informatics students want the *HomeQuiz* to be used more often, which shows the high acceptance of the instrument.

Students of course 2 (German literature) seem to be far more critical concerning the usefulness of the quiz. Despite the lacking representativeness of this questionnaire, students could be rather dismissive because of the missing link between learning objectives and quiz questions. Presumably, they realize that the quiz usage does not directly help them to achieve the learning objectives. Additionally, the results could strengthen the

assumption that quiz tools with the given answering formats are more useful in “hard sciences” or in courses with a high level of factual knowledge than in courses in which discourse and practical skills are focused.

The improvement proposals were rather useful for us. One student stated that a quiz should only cover one single topic. It could be discussed controversially, if this statement is didactically useful in itself. But the comment surely hints on the relevance and importance of formulating good questions. The wish to compare their result to other students gives useful hints: students are interested in their distance to the mean state of knowledge. Additionally, the publication of the results, e. g., in the following course sessions, gives the lecturers the possibility to comment on the state of knowledge of their students, to investigate problems in understanding (e. g., by asking further questions) and to give further explanations. Other benefits of the *HomeQuiz*, compared to the *MobileQuiz*, are the time saving towards performing a quiz during course time and the lecturer’s possibility to use the results for adapting the contents of the following lesson. The lecturer can analyze the *HomeQuiz* results before the lecture starts and realize the students’ problems in understanding. But in spite of these benefits, lecturers and students mostly appreciate the interruptions during a lesson as a didactic method for reactivating and motivating, which is irrelevant when only using the *HomeQuiz* functionality at home.

Despite the promising results, the value of the quiz still depends on the quality of the questions. Creating good questions is a considerable effort, which varies with the content’s nature. Additionally, the tool itself is constrained: for example, lecturers cannot determine if the number of votes corresponds to the number of participants, as one student is theoretically able to vote more than once as the participation is anonymous. The usage of the *HomeQuiz* raised the question again, whether the participation in the quiz should be graded or not. Taras [186] states that grading such tasks could undermine the main function of the quiz: students should be enabled to explore the contents, to make errors, and to identify their gaps in understanding without being afraid of worsening their grades. Therefore, the traditional educational rule has to be accepted here as well: the *HomeQuiz* (as all other tools in creating a learning environment) is an offer – students themselves are responsible for using it or not.

Evaluation of Different Mobile Scenario Parameters with the *MobileQuiz2*

With the implementation of the *MobileQuiz2*, a higher number of teaching scenarios could be analyzed with little effort. The lecturers did not need any programming skills anymore, which were previously required to change individual parameters in already existing scenarios or to create completely new ones. With that new possibility in designing the details of response scenarios, it was interesting to investigate the best use of the elements and parameters to support learning and motivation of the students.

A study was performed in the Fall semester 2014, in which twelve different learning scenarios were modeled within the *MobileQuiz2* to investigate the effects of different scenario parameters on the students’ acceptance. The study was part of the bachelor thesis of Licheng Yang [20] and published in Schön et al. [12]. The used scenarios were mostly multiple-choice feedback quizzes with differences in structure and appearance. Among other things, one of the scenarios investigated the differences between an open text answer format and predefined single choice answers. Following, only six parameters

were described in more detail, as the scenarios *font size* and *system performance* did not result in useful insights. Section 5.2.1 shows the investigated parameters and describes the main differences.

One of the tested scenarios contained one text answer, a progress bar, and a button to go forward. But the same topic could also be presented without a progress bar, with all questions and answer fields visible right from the beginning. Another scenario was a multiple-choice question with an image, whereas the alternative scenario used a mere textual description. The goal of the evaluations was to get a first qualitative feedback on those different scenarios to assess possible fields of further research. A focus was to get an impression on which parameters hold the potential to be rated as user-friendly and therefore support the students' learning behavior.

All participants of the study were enrolled in the course "Praktische Informatik I" which is taught every second semester at the University of Mannheim. Approximately 200 students attend the course's final exam. Most of them study business informatics (1st semester), business mathematics (5th semester) or business teaching (5th semester). The course comprises two weekly lectures, a weekly exercise and eight tutorials. The weekly exercise class lasts 90 minutes and has an estimated number of 40 to 60 participants. The eight weekly tutorials also last 90 minutes. The eight tutors are responsible for approximately 25 students in each tutorial. The attendance is not mandatory.

The exercise and the tutorials were used to test the different scenarios. To motivate the students to attend the scenarios the lecturer used questions and tasks of previous exams. The scenarios were created by implementing the quiz questions in compliance with the investigated parameters. Subsequently, the students took part in the scenarios. The exercise instructor or the tutors lead the discussion of the tasks. Altogether, five exercises and three tutorial days were investigated (all eight weekly tutorials took place on the same day of the week), with 15 minutes each for the case study. When the quiz was planned, ten minutes for the question session were scheduled since the students needed time to access the quiz website via the QR code. In the first exercise, the *MobileQuiz2*'s basic functionality was tested for the case study environment. After this test, the tool was configured so that it was ready for the real case study. At the beginning of the study, all students were informed that they would be participating anonymously in a study to investigate various parameters in scenarios, without letting them know which parameters were investigated in detail. The scenarios during the large exercises contained the same questions for all students because it would have been difficult to divide all participants in the same room into two large groups, as they would easily recognize the investigated parameter by comparing their scenario to the scenario of their neighbors. But the tutorials gave the perfect opportunity to divide the students into two groups as they do not meet during the tutorial time. Thus, it was possible to compare the results between the different groups. But the number of students in the tutorials varied slightly due to the voluntary participation. As a result, the two groups which we compared were not absolutely balanced. Moreover, the participants who showed up in the weekly tutorials were not steady. In order to balance the number of participants, the tutorials were arranged in a way that they formed two groups of almost the same size. The students of tutorial 1, 3, 5 and 7 formed group A, whereas the students of the tutorials 2, 4, 6 and 8 formed group B. The students were asked for their perception and opinion of each scenario directly after its usage.

TABLE 5.6: Selected parameters and hypothesis of the students' behavior.

#	Parameter	Description and Hypothesis
1	Graphical vs. Textual Content presentation	The same question is presented to students by either graphics or text. We assumed that students prefer a graphical presentation. Long texts would demotivate the students, and consequently decrease the task quality.
2	Multiple Choice with all true or false	The question is offered with several choices, where the choices are either all true or all false. We assumed that students will need substantially more time to answer all questions since their intuitive beliefs make them unsure.
3	Textual Answers with Special Characters	The questions have the same content, but the students have to answer them in the form of either text or JavaCode. We assume that answering with special characters will force the students to make a greater effort due to the limited keyboards.
4	Color	The application used a different color scheme for the background, text field, text, etc. We assumed that the students would be more motivated by pleasant colors to participate in the scenarios.
5	Progress Bar	Several questions in a list are either presented with or without a progress bar which showed the questions yet to come. We expected that students would be more impatient without a progress bar.
6	Grouped Content	All questions are not shown on the same page, but grouped according to the content. The students could navigate to target groups from the main menu. We assumed that through structural grouping the students would get a better orientation and thus answer the questions better.

Almost every student participating in the experiment also participated in the evaluation. About thirty statements per parameter could be collected. The handling of the prototype and the acceptance of the respective scenarios could be evaluated. Additionally, a qualitative user feedback was collected about the positive and negative aspects of every scenario with open questions at the end of the evaluation.

Parameter 1 - Graphical vs. textual content presentation: The quiz in this scenario asked questions about classes in the programming language JAVA. One group got a typical class diagram represented as a figure, whereas the other group got a description of the class relations. As shown in Figure 5.26 and Figure 5.27, there was not much difference in user experience between the two scenarios. We expected that the figure scenario would be favored, but about two thirds of the students rated both scenarios as “pleasant” or “very pleasant”. One third was not pleased with either scenario. As reasons they stated the good or bad clarity in both cases. The students were annoyed that they had to scroll up to the picture and down to the answers for every question. In the other case, they mentioned the pure amount of description text as too much

and visually unstructured. Interestingly, the group with the picture answered questions about the relationship between the classes more precisely whereas the other group was superior in questions about the class details.

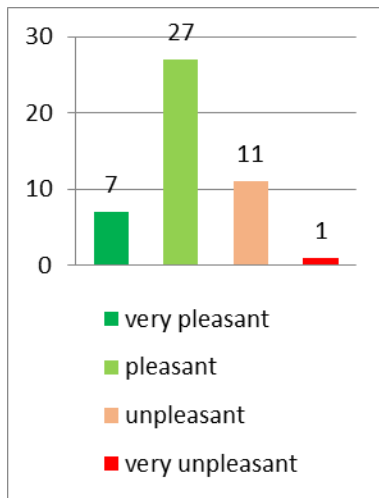


FIGURE 5.26: Students' satisfaction when using a quiz with an image.

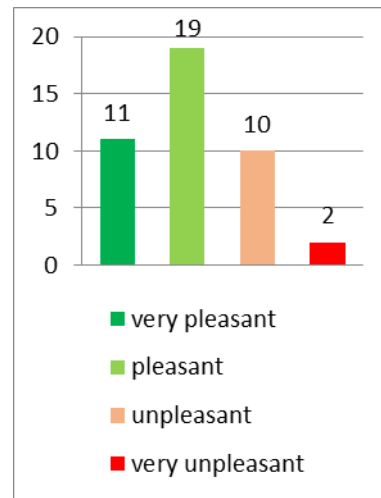


FIGURE 5.27: Students' satisfaction when using a quiz with text only.

Parameter 2 - Multiple Choice with only true or false choices: The scenario showed a picture and several multiple choice questions. The evaluation in Figure 5.28 shows a result similar to the first one. About two thirds of the students were pleased with the experience. As expected, they took more time to finish the questions compared to the first parameter test, and the level of noise in the classroom grew. The students mentioned that it should be announced explicitly when no choice is correct, as they were very confused with that possibility.

Parameter 3 - Textual answers with special characters: Figure 5.29 shows the results when students had to use source code in their textual answer. Source code contains many special characters and is more difficult to enter on a mobile phone. Usually, special characters like square brackets and the ampersand are on a second or third page of a mobile phone keypad and therefore not directly visible. But more than about the input of special characters, the students complained about the size of the input text field as they thought it was too small to keep an overview over open and closed brackets.

Parameter 4 - Striking color scheme: In this scenario we changed the color scheme of the quiz into bold green, blue and pink. The students showed mixed reactions. We assumed more severe protest but were surprised by the results. Beyond the fact that the look was not professional and some students noted that pink “was no serious color”, we also got many reactions stating that the bold colors supported a better overview of the quiz; more than half of the students liked the color change.

Parameter 5 - Progress bar: This scenario displayed only one multiple choice question and a “next” button on the page. With a click on this button the next question was

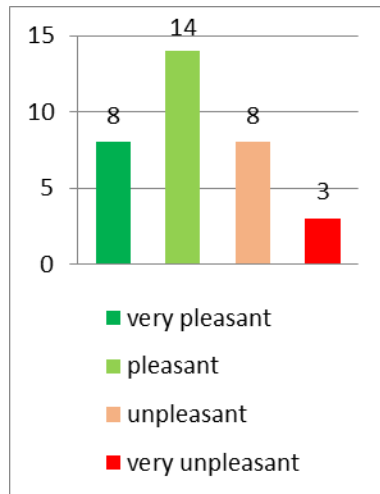


FIGURE 5.28: Students' satisfaction when using a quiz in which all answers were correct or wrong.

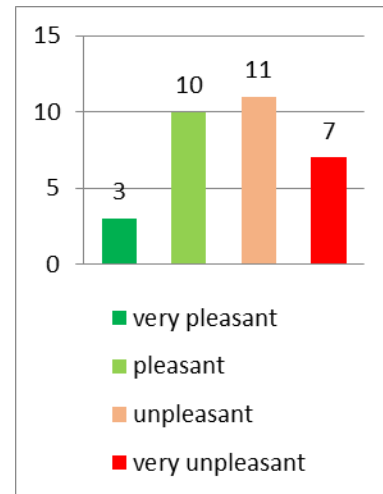


FIGURE 5.29: Students' satisfaction when they have to enter special characters.

shown. One group also got an additional progress bar, showing the amount of questions yet to come. Figure 5.30 and Figure 5.31 shows a similar positive feedback of both groups. They appreciated the increased clarity of the quiz with only one question at a time. Students without the progress bar claimed that they wanted to see the number of remaining questions. Both groups asked for a “back” button as it was not possible to return to a previous question.

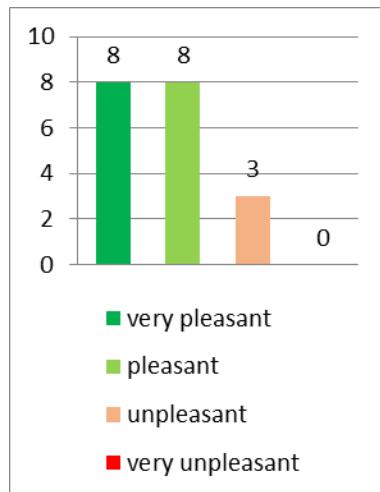


FIGURE 5.30: Students' satisfaction in a quiz with an additional progress bar.

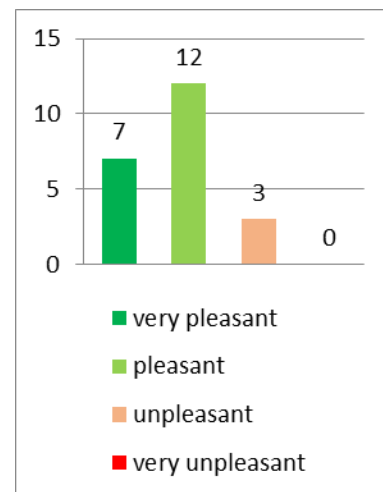


FIGURE 5.31: Students' satisfaction in a quiz without a progress bar.

Parameter 6 - Grouped content with index: In preparation of the upcoming exam, this scenario covered all topics of the course. About five questions for every topic were grouped together, and the students could choose between the topics via an additional index page. After completing a topic, they could return to the index page and go to another set of questions. The overall submit button was not available until at

least one topic was completely answered. Students really liked this setup in regard of the upcoming exam, as everyone could attend to his/her favored field.

As digital natives, today's students are confronted with a wide range of website and application designs. On one hand, this lets them easily handle the use of QR codes and mobile applications, on the other, it makes them very sophisticated in the evaluation of user-friendliness and design aspects. As students are used to professional products in applications and web pages, they are, for example, very impatient concerning the loading times. During the study, the most basic requirement was the system's performance. Students mentioned the performance as important in every scenario, especially in scenarios with a bad performance due to a high network delay.

Besides, students are used to so many variations of applications that they quickly adapted to concentrate on the contents first. Therefore, the lacking outrage on color variations is comprehensible. They focused on a clear overview of the required task. They wanted to quickly recognize important content information and the click-able elements. Following the usability premise: "Don't make me think!" (also see Krug [127]), they did not want to have to explore the functionality of the tool first. Thus, they preferred layouts supporting clearness, including colors with a high contrast. They also wanted to overlook their range of possibilities as quickly as possible and did not want to scroll the page to discover more answer possibilities, respectively scrolling back up to inspect the given information more deeply. They also liked the scenarios with only one visible question, where they could navigate to the next question with a forward button or an index page. The usage of a progress bar further supports the overall overview, but only with a small margin. Using figures to visualize a problem may not be supportive in every case. Especially when asking complex and detailed questions, students often prefer a textual problem description with textual structure elements like *bold* or *italic*. As tested with the second parameter, students got irritated when the known elements were not used in the usual way. When none (or every) given choice was correct, they became uncertain and started to discuss this fact with their neighbors. Concerning the limitations of our study, students were not used to a meta-analysis of their learning material, especially when the analysis focused on design differences. Students were told about the purpose of the meta-analysis, but surely had problems describing their perception on minor design differences. Additionally, students could not directly compare the parameter variations, e.g., having a progress bar versus not having a progress bar.

Customized Scenarios, Anonymous Participation and Installation

After the evaluation of parameters of learning scenarios and establishing the *Mobile-Quiz2* in the ongoing lectures, the students were confronted with a number of different teaching scenarios with an increasing complexity. Lecturers modified their scenarios on a weekly basis and also designed completely new scenarios. They typically started with a basic quiz and customized the amount of visible questions and answers. This finally resulted in the "InputPhases"-scenario. When using the "MessageBoard", the adjustments were not as much in the scenario's logic, but in the additional prevention of students trolling and the way the scenario was integrated into the lecture. At start, it ran parallel to the lecturer's usual teaching and was later displayed only at selected points in the lecture. This process was analyzed with a short evaluation at the end of the Fall semester 2015 (full survey can be found in Figure E.5). A survey scenario

was designed in the *MobileQuiz2* application, and a questionnaire for students and another one for lecturers was created. Although more than a dozen lecturers had been introduced to the *MobileQuiz2* by then, only a few actually used it regularly during the ongoing semester. Most lecturers wanted to wait for the next semester, using the vacation period to get used to the system and prepare their scenarios. Therefore, only three questionnaires were created in total, and the hesitating lecturers and the performing lecturers were interviewed separately. The survey was conducted with all lecturers who had already received an introduction to the new application (users and hesitators). The students of a course which used a high variation of scenarios throughout the semester were interviewed separately. The results of these surveys were originally published in Schön et al. [13].

The first survey was conducted in an exercise of informatics. It had 27 participants. The students were confronted with seven statements and gave their rate of agreement on a four-point Likert scale ranging from “absolutely disagree” to “absolutely agree”. Four-point scales were used in this test to force the students to tend in one direction as there is no neutral middle. The results showed that students mostly enjoyed the new quizzes. When asked about the improvement in understanding the course contents, the majority of the students stated that the quizzes supported their understanding. Figure 5.32 shows that more than 50% agreed with the statement. But at least 30% of the students did not think that their understanding was supported by the scenarios. The students were also asked if the system distracted them more than it helped their understanding.

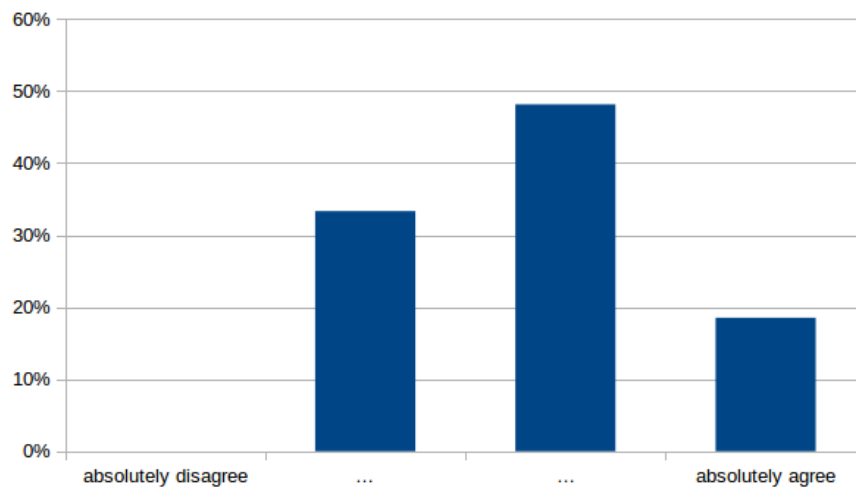


FIGURE 5.32: Students’ answers to the question if the *MobileQuiz2* helps to understand the course content.

The course’s lecturer performed at least five different scenarios during this semester, including several variations of a quiz, a “MessageBoard” and two “InputPhases” scenarios. The students were asked if the variations in teaching scenarios increased the course’s attractiveness. Figure 5.33 shows that more than 80% of the students stated, that these variations made the course more interesting. When asked if the anonymous participation was important, 80% of the students agreed with that statement. Less than 20% slightly disagreed with this statement, as shown in Figure 5.34. The amount of “absolute agreement” is very high with about 60%.

Figure 5.35 shows the results when asked if the students would accept to install separate software to participate in the quizzes. The answers were not as explicit as with the

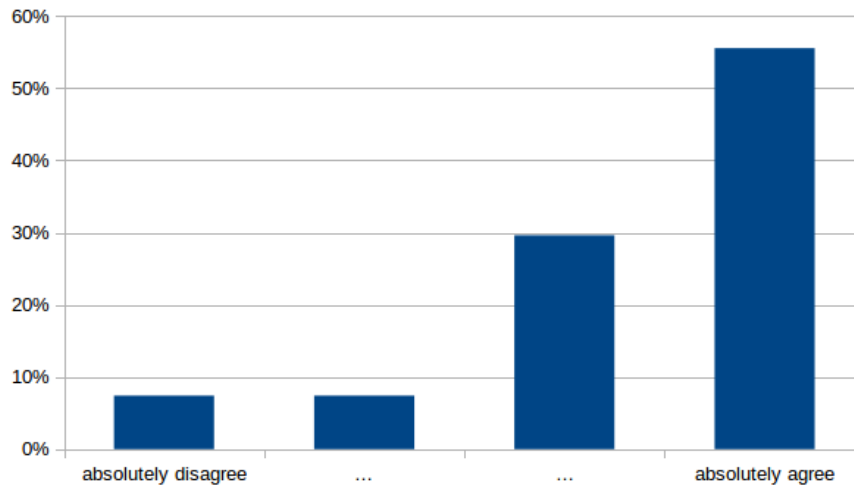


FIGURE 5.33: Students' answers to the question if the new variation makes the course more interesting.

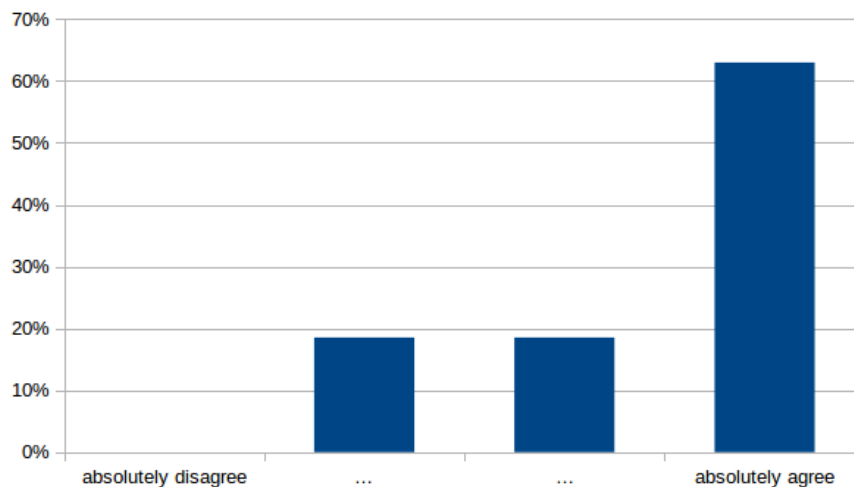


FIGURE 5.34: Students' statements to the importance of an anonymous quiz participation.

previous statements. About 60% of the students would not accept to install additional software. Only 10% stated that they “absolutely agree” to accept installing additional software on their phone.

The students were also asked if they want to be graded with the help of these quizzes. Figure 5.36 shows that almost 80% “disagree” with that and almost 50% even “absolutely disagree”. Finally, the students had the opportunity for free comments. They mainly noted that the stability of the system has a high priority to them.

The results gave an impression about the students' priorities. Despite appreciating the new variations in teaching scenarios, they had concerns. Students liked the varying quizzes and thought that they supported their learning process, but they did not want to be graded with these quizzes and wanted to stay anonymous. These findings supported the initial didactic focus of the tool: students should be able to apply knowledge in an active way and lecturers should thereby be able to identify gaps in understanding. A non-anonymous or even graded usage would lead to pressure during the learning

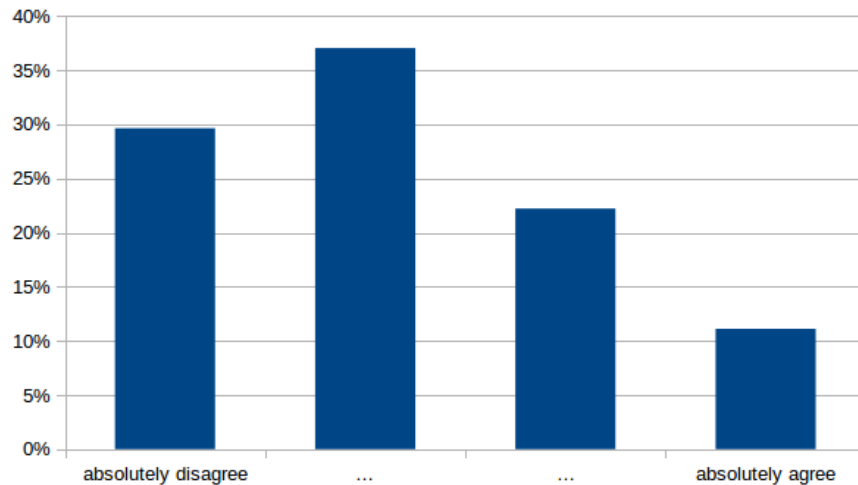


FIGURE 5.35: Students' statements to the acceptance of installing a separate software.

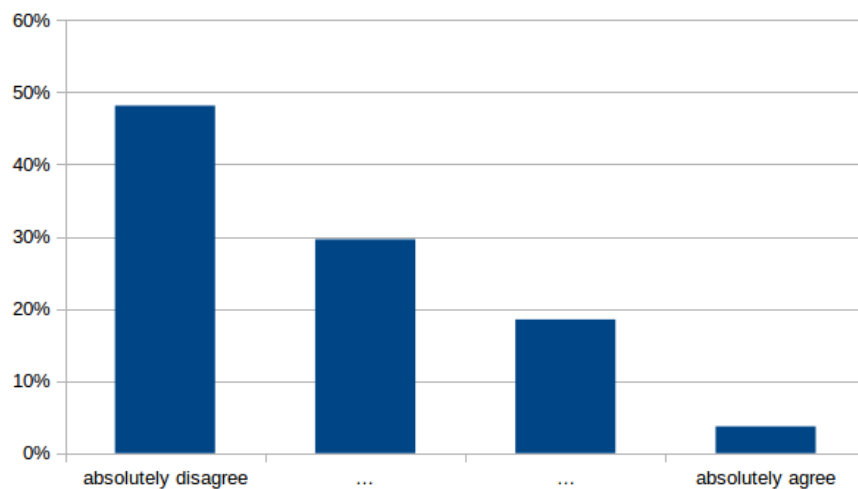


FIGURE 5.36: Students' statements to the acceptance of being graded by the system.

process. Consequently, only smart students would give answers and therefore make a realistic analysis of the students' understanding impossible. Weaker students would hold off and would not benefit of the potentials of the tool to identify their own strengths and weaknesses. Nevertheless, there could be additional reasons why students refused to be graded by the ARS: maybe they feared the system's sensitivity for manipulation, or they preferred assessment formats in which they could demonstrate their knowledge with more than one multiple choice answer (as the input of longer free-text answers on mobile devices is very uncomfortable). Students do not want to install additional software either. So, implementing a native application, which could enhance performance and increase the functionality would cost at least a bit of the students' acceptance.

5.2.2 Acceptance by Lecturers

Regarding the implementations' acceptance, the opinions of the lecturers were as important as the acceptance by the students. Before the students came in contact with the quizzes, the lecturers had to be convinced that using quizzes will improve their courses.

To get a better impression of the lecturers' opinions, several surveys and interviews were performed. Because of the relatively small number of available lecturers, the evaluations were mostly done in face-to-face interviews which did not have a quantitative goal. Instead, they were focused on a qualitative feedback to improve our models and their implementations.

Experiences during the Pre-Study

During the first experiments with the *MobileQuiz* (published in Schön et al. [6]), no quantitative evaluation on the acceptance of the *MobileQuiz* was done from the lecturers' point of view. But the qualitative feedbacks were quite positive. The integration into the existing university's e-learning system - which most of the lecturers used - simplified the administration of the quiz rounds. Lecturers were not forced to learn a new system and were directly able to use the application. Furthermore, using the students' devices reduced preparation and post processing time. The lecturers appreciated that no additional time was needed for distributing and collecting the voting devices. Although this minimal technical preparation effort would have enabled spontaneous quiz rounds, lecturers only used quizzes which they prepared prior to the lecture. Nobody created a spontaneous quiz during lecture time.

Lecturers Experiences with the *MobileQuiz*

During the first semesters in which the *MobileQuiz* was used, an evaluation with the lecturers was performed, as published in Klinger et al. [2]. The experiences and opinions were evaluated separately. Seven lecturers from five different departments were questioned. At the time of the evaluation, they had between one and ten weeks of experience with the usage of the *MobileQuiz*. Our main interest was the contentual and technical efforts in the preparation and execution of the quizzes, as well as an estimation of the students' increase in motivation and comprehension (the original survey data can be found in Appendix D).

Almost all questioned lecturers said that the technical effort in preparation and execution of quizzes is "small" or "very small". Only one lecturer stated the quiz's execution as to be complex. Indeed, two lecturers stated that the effort to create suitable questions consumes much preparation time. The others stated the contentual effort during the preparation to be "small" or "very small". All lecturers had the impression that the students easily handle the used technology. The lecturers stated that they did not think it was a disadvantage that not every student owned a mobile phone. Students without the possibility to join the quizzes directly teamed up with their neighbors and discussed the questions together. This observation is confirmed by other researches, in which the teamwork of several students on one device isn't seen as a disadvantage neither (e.g., [82]). In some lectures, quiz applications are knowingly used with interactive partner and group discussions [175]. By this form of collaborative learning, discussions are encouraged during the lecture. Additionally, students who only watch their fellows in participating in the quizzes perceive the lecture as less boring and more interactive than lectures without a quiz [192].

Regarding the temporal workload, which is caused by using the *MobileQuiz* and the discussion of the results, lecturers gave different answers. The quiz was used once during

an 90-minute lecture in most of the cases and took about ten minutes, depending on the amount and complexity of the questions. The reason for this long period mostly lay in technical problems of individual students. It could be observed that this time decreased during the ongoing semester, as the students got used to the process. The presentation of the results and the discussion with the students took about five minutes. But when the results weren't as expected, the discussion could take noticeably longer. Together, the execution of the quiz and discussion of the results took about 15 minutes, which had to be scheduled beforehand. However, all seven questioned lecturers said that this investment was justified.

The following quotation of a lecturer in German studies is representative:

“Ich bin der Meinung, die Studierenden nehmen aus dem Quiz viel mit, da sie persönlich betroffen sind. Ich glaube, der Zeitaufwand nützt denen, die falsch geantwortet haben, weil sie etwas lernen und denen, die richtig geantwortet haben, weil sie sich über ihr Wissen sichtbar freuen. Wenn ich statt Quiz 15 Minuten länger vortrage, hören die Studierenden irgendwann nicht mehr zu. Mit dem Quiz werden sie aber aktiviert und sie interessieren sich für die Ergebnisse im Gegensatz zu meinem Dauermonolog. Im aktuellen Semester fragen die Studierenden auch dauernd, wann wir wieder ein Quiz machen.”

The English translation is:

“I think, the students take a lot out of the quiz, because they are involved individually. I think the time exposure is beneficial for the students who answered incorrectly, because they learn; and it is beneficial for those who answered correctly as they are visibly happy about their knowledge. If I talked 15 minutes longer instead of performing a quiz, the students stop listening at some point. But the quiz activates them, and they are interested in the results in contrast to my permanent monologue. During the current semester [one semester after the first usage], the students ask permanently when the next quiz will be performed.”

Lecturer Evaluation of the *HomeQuiz* Variation

The *HomeQuiz* variation of the *MobileQuiz* was evaluated separately (see Schön et al. [8]). The two lecturers of the two investigated courses were interviewed again. In the opinion of the lecturer in Advanced Computer Networks (ACN), the most important benefit of quizzes is the feedback about how well a topic is understood by his students. This is especially useful when the quiz is used in its *HomeQuiz* variation in combination with exercise sheets, as difficult questions can be discussed and explained in the following lesson. It also seems to be important that the questions of the quiz match well with the content of the current exercises. The students of ACN were not forced to prepare the homework exercises. Therefore, the quiz especially helps to motivate some of the more passive ones. The additional effort for inventing good questions is seen as a time-consuming activity. But it is also seen as decreasing in the future because of the re-usability of already existing questions.

The lecturer of German literature described the *HomeQuiz* as a useful instrument because it combines the advantages of classical e-learning with the advantages of the regular *MobileQuiz*: as e-learning itself is traditionally independent of time and space and gives immediate, teacher-independent feedback, the *HomeQuiz* offers the lecturer the opportunity to constrain the timeline and to give personal feedback in the following course session. The lecturer can then look at the aggregated results before his next lesson and discuss them together with his students. The “single-player” mode of the quiz itself is complemented by interactivity with the teacher and other students. The teacher can then explain detected understanding deficits in personal or group discussions. The lecturer further stated that the *HomeQuiz* could not fully meet the requirements for his seminar, as the format of the quiz was not appropriate concerning the learning objectives of the course. Students are supposed to practice practical skills, and therefore a one-dimensional answering format is not useful, even if some basic and elementary principles can be formulated in single and multiple choice formats. He summarized that the *HomeQuiz* is a “nice tool” that can make teaching more interesting at least.

Actual Usage of the *MobileQuiz*

The *MobileQuiz* was prototyped at the end of the year 2011 and tested during the following semester. The first lecturers who used the *MobileQuiz* during their courses were from the area of informatics and familiar with predecessor applications. In preparation of the Fall semester 2012, the didactic department of the University of Mannheim promoted the *MobileQuiz* and provided personal introductions to its functionality. Figure 5.37 shows the number of *MobileQuiz* objects used in ILIAS courses since then (ILIAS handles the *MobileQuiz* as learning object like). In the Spring semester 2013, the usage grew from 28 to 37 quizzes, and up to 55 in the Fall semester 2013. After a small drop in the following Spring semester it reached a usage of 80 quizzes in the Fall semester 2014. The number decreased again in the next Spring semester and reached 71 quizzes in the Fall semester 2015.

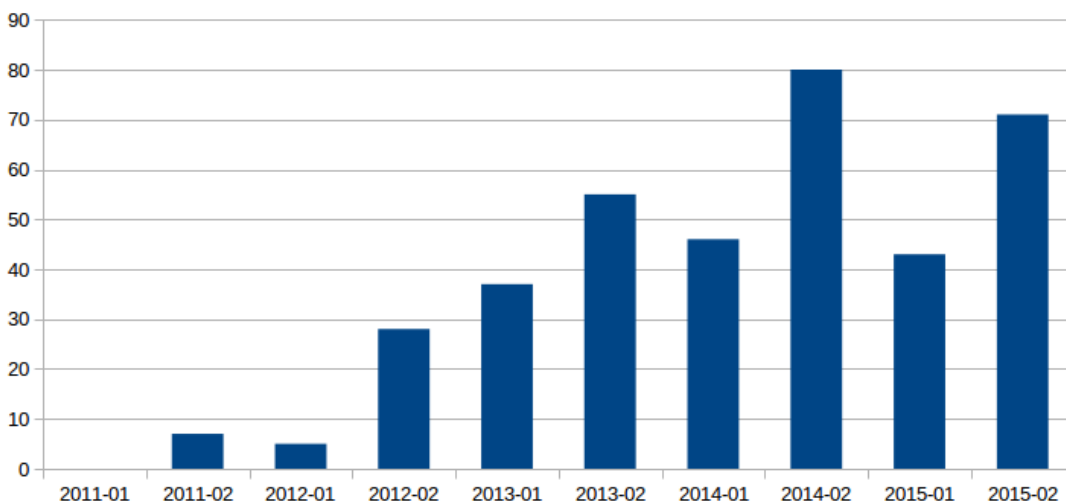


FIGURE 5.37: Number of *MobileQuiz* objects used in courses in ILIAS since its implementation.

Considering that some of the pioneer users of the *MobileQuiz* switched to the *MobileQuiz2* during the Fall semester 2015 for some of their courses, the amount of all used

quizzes still increased. But it is visible that the rate of increase calms down steadily. The tools aren't promoted as heavily as in the beginning anymore, and the highly interested and motivated lecturers already use the quizzes. But as the number of *MobileQuiz* usages can easily be tracked within the university's learning management system ILIAS, the amount of used external ARS applications, which are used by some lecturers, cannot be counted directly. The numbers show that the quiz has been adapted by a group of lecturers and is used regularly. Furthermore, as it is an official ILIAS plug-in, it is fully supported by the university's IT department and published under the GNU General Public License to be used by the entire ILIAS community.

Evaluation of the Acceptance of the *MobileQuiz2*

As explained, with the launch of the new *MobileQuiz2*, the lecturers were able to individually modify their teaching scenarios to perform more specialized quiz rounds in their lectures. To evaluate the concepts and performance of the *MobileQuiz2* a short survey was conducted to question the lecturers who were introduced to the *MobileQuiz2* or were already using it (the results were originally published in Schön et al. [13]). The survey was directly written within the *MobileQuiz2* as a special survey scenario and answered by the lecturers like a usual *MobileQuiz2* quiz round (the original survey data can be found in Appendix E).

Six lecturers from different disciplines were asked to give a qualitative feedback. Two of them had an informatics background whereas the others came from didactics, mathematics and economics. When asked about their commitment to teaching, most stated that they were very committed to it (with one saying he was only slightly committed). But not every lecturer had already used the quiz in the past semester, even if every one of them showed high interest in the system and took part in a personal introduction. When asked for the reasons, the answers were mainly that they had time constraints during this phase of the semester and that the current course concept does not include the quizzes yet. But they were looking forward to the next semester; they intended to use the semester break to prepare for the next courses. When asked how the students could benefit from the customized scenarios, the answers were similar to the benefits of regular ARSs. They mentioned the students' reflection of their own learning process and the lecturer's insight in the students' knowledge base. All lecturers said that the quiz tool could be used after an introduction without further support, but the greater challenges were to get all the students to participate, the creation of suitable quiz questions and the design of good teaching scenarios. Considering recommendations for improvements, the lecturers wanted a better usability and overall stability of the system to support large user groups. They also wanted to see more predefined teaching scenario blueprints. After being introduced to it, lecturers enjoyed the new application and thought it was easy to use. Nonetheless, only a few lecturers already used the system and found the time to create suitable content and to integrate it into their courses. The majority of the more complex scenarios were created by only one lecturer. The other lecturers used already known questioning scenarios or the few scenarios they had already created before the courses started. One explanation could be that the system was still too complex to be easily used by the lecturers on their own, even after an introduction. The improvement to an even more intuitive usability and the offer of a permanent didactic support could help to overcome these challenges. Didactic support is also crucial to solve the problems of getting students to participate, of creating suitable quiz questions, and of designing good teaching scenarios. Concerning the latter, one advantage of the system is the

possibility to access teaching scenarios that were created by other lecturers. Thereby, teachers can get inspirations for their own courses. As expected, all lecturers that answered the survey described themselves as “very committed” or “slightly committed” to teaching. To widely spread the usage of innovations in teaching, another challenge would be to convince less committed lecturers. To achieve this, convincing arguments and a user-friendly interface are essential elements.

Evaluation of a Graphical Scenario Editor for the *MobileQuiz2*

In the early stage of the *MobileQuiz2* implementation, the scenario blueprints had to be defined with a text editor in an XML file. Creating and modifying scenarios was already possible for lecturers themselves, but it required some basic computer science skills. Therefore, a graphical scenario editor was developed, implemented and evaluated in cooperation with Thilo Weigold [19]. The following results were generated during his work.

The evaluation of the graphical scenario editor was performed with four test persons, who already used the *MobileQuiz2* and volunteered for the interviews. Three of them had an equal profile, as they were research assistants in the area of computer science. The fourth participant was from the area of didactics and was actively involved in the *MobileQuiz2*’s development process. After a brief introduction to the functionalities of the new editor, the participants had a few minutes to get a first impression of the system. A part of the interviews consisted of three basic tasks, which had to be accomplished by the test persons themselves. These tasks involved the implementation of three learning scenarios, which varied in their level of complexity and were supposed to acquaint the subjects to the new editor. The completion of the tasks required the usage of some of the editor’s basic features and concepts. The first task was designed to introduce the basic mechanisms to build *MobileQuiz2* scenarios. At this point, the usage of predefined components was crucial. The second scenario was supposed to introduce typical quiz components by defining a scenario which contained a small introductory text and a single choice question. The focus of the third task was the usage of default components and the introduction to rules. The scenario was supposed to define a quiz, which included an introduction text, a picture and an open question. Additionally a small feedback message should appear after the student submitted an answer. After completing these tasks, the test persons were asked to rate specific aspects and functionalities of the authoring tool within a questionnaire. It used five-point Likert-scale questions to record the subjective assessments. At the end of the interview the lecturers were able to provide a personal feedback and suggestions.

One intention of the lecturers’ tasks was to test how they dealt with the various features and functionalities. Another one was to observe how the scenario’s complexity influenced their ability to use the editor. The usage of predefined components was understood quite fast and was rated as easy. The second and third task required the usage of default components, attributes and rules. Especially the differences between objects and attributes was not transparent to the lecturer right from the beginning. They had difficulties to concretely use them to create a simple introductory text for the intended quiz. At this point the lecturers mentioned that the distinction of a scenario and a concrete entity of a quiz was not obvious. Therefore, they did not understand why the order of objects in the scenario structure did not matter in this phase. Task three was rated as quite complex due to the implementation of the feedback functionality. All participants had

difficulties to define an action element for the rule that set the corresponding attribute of the feedback text to “visible”. It was stated that the definitions of the action’s operators were not evident enough. No one accomplished the third task without support of the interviewer.

TABLE 5.7: Average answer values of the survey with four lecturers about the graphical scenario editor (if not stated otherwise: 1=“I fully disagree”, 5=“I fully agree”).

Question	Average value of answers
After finishing the three tasks, how would you rate the learnability of the authoring tool? [1=“very low”, 5=“very high”]	4.25
The authoring tool is easy to use.	3.25
The layout of the authoring tool is clear.	3.25
The interaction with the authoring tool is responsive.	3.50
The authoring tool and its functionalities are useful.	4.75
After testing the authoring tool I feel [1=“not very convinced”, 5=“very convinced”]	3.75
I can imagine to continue using the tool for creating learning scenarios.	4.75

In the second part of the interviews the lecturers had to rate the overall authoring tool with a questionnaire. The gathered values were supposed to show tendencies of specific aspects, like learnability, ease of use and a perceived usefulness of the authoring tool. The results of the questionnaire are summarized in Table 5.7. The overall experience with the tool was rated as fairly well. Although the test persons experienced the way of creating a scenario as quite complex, most of them had the feeling that they would learn and understand the concepts of the authoring tool by routine.

The results show that the authoring tool was moderately well rated in terms of ease of use, clear layout and responsiveness. Lecturers mentioned that the representation of the scenario’s structure with its objects and attributes appeared overloaded and confusing when looking at it for the first time. Furthermore the naming of operators and properties was not obvious. Despite those lacks, the perceived usefulness was rated with an average value of 4.75 out of 5. Most of the participants could imagine to continue using the tool in future.

Despite a fairly well and positive overall rating, the usability of the authoring tool was limited due to its abstract structure of the underlying learning scenarios. The *MobileQuiz2*’s generic and abstract model limited the usability for users, who were not familiar with the *MobileQuiz2*’s abstract concepts. A trade-off between usability and complexity was inevitable. But the question of which direction is more compatible for the usage of an authoring tool was not treated any further. Nevertheless, it could be recognized that the participants rated the capability to configure and create a scenario without restrictions as very significant and mentioned that it should be maintained. Furthermore, usability improvements like layout, representation and various graphical elements had to be improved and simplified. The integration of predefined components proved to be quite beneficial and facilitated the construction of new learning scenarios for the test persons. It was mentioned that more predefined components should be added.

5.3 Discussion of Results

The evaluations show that the presented approaches are suitable. The technologies are capable of serving the requirements, students accept the interruption of lectures, and lecturers are willing to adapt the new opportunities.

5.3.1 Technical Aspects

The results of the experiments with the QR code described in Section 5.1 were confirmed by further quiz usages in following lectures. Students are quite familiar with the technology and have very little trouble to use it during lectures. QR codes are proven and easy to use, if they are displayed in an appropriate size, the error correction level is low, and a suitable amount of letters are encoded. But nonetheless, some students still join the quiz rounds manually via the URL in almost every classroom usage.

The level of stress on the network is acceptable, too. The results show that a WiFi network is capable of handling a classroom full of devices. This will even be improved in the future, as the increasing propagation of mobile devices forces the universities to upgrade their network infrastructure. The question at the beginning of this thesis was *if* all students own a suitable mobile device, whereas the question now shifted to *how many* mobile devices are used by a single student.

The distribution of different operation systems and brands of the students' devices showed that no technology dominates. The focus on brand-independent technologies is therefore preferred when considering quiz applications.

Whereas the basic *MobileQuiz* had no performance issues at all, the advanced *MobileQuiz2* had problems with complex scenarios. The amount of database objects in the *MobileQuiz* were quite reasonable, compared to the many database entries for the *MobileQuiz2*.

5.3.2 Students' Point of View

The evaluations with students showed that the QR code is a suitable way to accelerate the access to quizzes. The *MobileQuiz* is rated as easy to use in general, whereas the opinion about the *MobileQuiz2* depends on the respective teaching scenario and its performance.

In every presented survey, the majority of the students stated that their interest in the lecture and its contents was raised when a quiz was used. These results were even more obvious in large lectures. In contrast to the interest, the students' motivation did not increase that well with the usage of quizzes. Whereas some students absolutely agreed to a raise in motivation, others absolutely disagreed. No difference between large and small lectures could be recognized. Except the survey about the *HomeQuiz*, every survey showed that quizzes had no significant influence in the motivation to learn the course contents. The results when asked if the quizzes support the understanding of the course contents were ambiguous again. The majority of students in the large economics lecture stated, that the quizzes supported their understanding at least a bit. In contrast, the results of the seminar of German literature showed no significant result. The reasons

could be the number of students per lecture, or the different types of content. The evaluation with a control group in the course of German literature also showed that a direct didactic benefit is difficult to recognize, although at least half of the class stated that they liked the quizzes. The grades of the two groups showed no significant difference. A more positive conclusion can be given about the *HomeQuiz* feature, which is mainly used for quizzes in preparation of lectures. The Students state that these quizzes increased their motivation and interest for the course preparation.

An evaluation with a different layout showed that students especially want a smooth quiz performance and a clear overview about the current task. This zero tolerance attitude for delays also covers the findings of Gannod et al. [92] and Keough [119], who detected the high aversion of students regarding performance issues. As Bouch et al. [35] confirms, when Web-applications do not meet the demands in delay, users get a poor impression of the whole underlying business. This could have an impact on the rating of the whole application and even the quality of the lecture itself. Regarding the overview, students want an obvious layout of the elements, which are an informational text or graphic, which are interact-able buttons, and which information belongs to which interaction element. Students therefore prefer a grouped content with one question per page, and dislike scrolling. The presentation of information with images and the presence of a progress bar can support this experience, but must not interfere with the overall overview. In addition, when performing established scenarios like knowledge tests, the various elements should be designed as students expect them to be. E. g., when each or none of the given choices is correct, students get irritated and start a discussion. But if a discussion is the intended target, this could be a suitable method (e. g., Caldwell [42]). The evaluation of the customized teaching scenarios showed that a diversion of suitable teaching scenarios can increase the students' interest in the course. But these have to perform smoothly, without performance issues, and thereby should not be confusing.

In summary, a great majority of students accepts quizzes in their lectures and want to use them regularly. The perceived benefits differ in courses of different sizes and topics. But the results also show that students do not want to install additional software to enable their participation in quizzes. Furthermore, they do not want to be graded and they want to stay anonymous. The impact on the students' learning success was not measured, as it was not a main focus of our evaluations. The measurement of students' learning success is a complex pedagogical problem which is subject of research in the pedagogical sciences. For an adequate amount of data, it may be necessary to collect the students' grades over a sufficient period of time. This comes with data privacy issues as grades are critical personal data. The experiment has to consider a significant pedagogical effort to exclude unwanted didactic side effects. Furthermore, a suitable reference group is needed, which may not be handicapped by not using the evaluated methods and tools. In consideration of the pedagogical difficulties, most of our evaluations were done in cooperation with the university's center for educational development, especially Melanie Klinger.

5.3.3 Lecturers' Point of View

The evaluations show, that the chosen technology environment fits the lecturers' needs. The inclusion of the quiz application into the existing learning management system spares lecturers the effort to learn an additional application. The use of the students' devices spares the additional distribution of clicker devices as the availability of suitable

devices among students is very good. The QR code is an easy-to-use method of access to the quizzes, which does not require complex explanations. Lecturers appreciate the honest feedback of the student's knowledge and thoughts, which they are able to collect with knowledge quizzes. Regarding this aspect, *MobileQuiz* and *MobileQuiz2* do not differ from other existing ARSs. The opportunity to create novel teaching scenarios and individually customize existing ones is a remarkable benefit in comparison to other systems. Lecturers like the possibility to customize the quizzes exactly to their teaching methods. But they do not exploit the full potential by themselves. They need a set of preset examples, which offer a predefined choice and inspiration, so that they could pick the scenario which serves them best. They will not start to customize the scenarios or develop novel ones until they get used to the process and the application. Thereby, the support by experienced educational specialists is recommended to avoid situations where students exploit a given scenario to sabotage teaching. This was observed by Holzer et al. [107], too, who used a scenario quite similar to the "Message Board" of Section 4.3.4. When lecturers finally began to actively design new teaching scenarios, it was quite challenging to adapt the necessary knowledge and technologies. Improving this customization by providing a graphical scenario editor instead of the direct manipulation of XML files supported the lecturers in the task of individual customization, but failed to be sufficient. Lecturers still had to be introduced and trained for the editor and needed at least basic programming skills. Offering predefined scenario templates can be a further method to decrease the barrier of individual scenario customization. In addition to these technological challenges, the effort to fill teaching scenarios with didactically useful content could not be reduced. Developing sufficient teaching content for the quizzes still remains a large issue for the lecturers. But the steady use of *MobileQuiz* quizzes during the last semesters and the high interest in the *MobileQuiz2*'s features show that the lecturers see the benefits of quizzes and are willing to use them.

Chapter 6

Conclusion and Outlook

6.1 Conclusion

Didactic research showed that Audience Response Systems (ARS) are able to activate students during long lectures and increase their motivation and interest for the course's contents. But above all, ARSs support the students' ability to self-reflect their base of knowledge and identify their knowledge gaps. The availability of mobile devices among students offered the perfect opportunity to introduce a lightweight ARS to university lectures without having the efforts of distributing and recollecting special hardware devices. The *MobileQuiz*'s approach to implement a lightweight system into the existing university's learning management system, together with using the students' mobile devices is a suitable solution. Using a URL in combination with a QR code as access method showed to be a good solution which is accepted by students and lecturers. But with the increasing familiarity with such systems, the lecturers' requirements raises, too. They want to configure teaching scenarios to their individual needs, so that they are able to use them in their teaching style. Therefore, the *MobileQuiz2* was implemented. The *MobileQuiz2* is a novel ARS focused on offering a large variety of different teaching scenarios with a high amount of customization and flexibility within a single system. After the implementation, both applications were evaluated in real university lectures of different sizes by students and lecturers. The evaluations showed that the approaches fit well to the current requirements. They are able to increase a lecture's didactic value. But nonetheless, depending on the complexity of the solution, the technical implementation faces some serious bottlenecks. An actual increase in the students' learning success was not measured, as it was not a main focus of our evaluations. The measurement of students' learning success is a complex pedagogical problem which is subject of research in the pedagogical sciences.

The following sections present a detailed conclusion of the basic *Lightweight ARS* approach of the *MobileQuiz* and the advanced *Customizable Teaching Scenario* approach of the *MobileQuiz2*.

6.1.1 Lightweight ARS

In a first approach an electronic quiz system called *MobileQuiz* was designed and implemented, which can be operated via the university's e-learning platform ILIAS. Students

were able to use their own Web-enabled mobile devices, e.g., smartphones, to answer the quiz questions. The experience showed that more than 90% of the students owned a suitable device for participating in such quizzes. A QR code was used to ease the initial connection to the quiz server by encoding a URL on the quiz page. The QR code's readability was evaluated in a typical lecture room setting, and the impact of the *MobileQuiz* on the university's WiFi infrastructure was measured. Both evaluations showed that the design was well feasible in practice as the QR code showed a good readability, and the WiFi network was not overly stressed. The approach was further extended by the possibility to add multimedia content like pictures and videos to the questions and answers. But complex HTML5 technologies (3D objects and videos) should not be used in common lectures, because most mobile browsers still have difficulties to support them. The *MobileQuiz* only required an Internet connection and visual sight of the QR code (or the URL). Therefore, it was usable in remote video lectures or in a combination of remote and local lectures without any additional effort for the teacher.

The user evaluations and experiences showed that lecturers and students rated the system quite well and are willing to use it. The majority of students stated that the quiz usages increased their motivation, interest and understanding of the lecture contents. They want to use such kind of quizzes more often within their lectures. The number of quiz objects created in the university's learning management system show that lecturers recognize the benefits of these quizzes and use them regularly, even without an additional external motivation. But only a minority of lecturers that were not used to ARSs before are going to use quizzes for the first time without an external push.

The *HomeQuiz* extension has proven to be a useful instrument for interactive self-assessment. It allowed students to examine their gaps in knowledge individually from home and gave the lecturers the opportunity to react accordingly in preparation of the next lecture. Students stated that they were willing to ask more questions during the course, and they had a higher motivation to prepare for the courses. Fast accessibility and easy arrangement within the existing learning management system were important prerequisites for this result. But despite the time saving by collecting the students' state of knowledge before the lesson (and not during it), home self-assessment should be seen as addition, not as a replacement to an ARS, as not all didactic benefits of using an ARS during a lecture can be reached.

In conclusion, the lightweight ARS solution is a significant benefit to a university lecture and can increase the overall experience. But the tool did not improve the teaching by itself. It has to be embedded into an elaborated learning environment. One special challenge is the creation of useful and effective questions, which depends heavily on the nature of the topic. The development and implementation of the quiz has to be followed by support and advice for the users.

6.1.2 Customizable Teaching Scenarios

With the *MobileQuiz2*, a novel model for describing different learning scenarios in typical classroom settings was developed. By focusing on objects, attributes and rules, it became possible to describe a large number of different scenarios with one single model. A prototype application was implemented based on that model, which was introduced to several lecturers of different departments. Using that prototype, a general evaluation of the concept was performed, as well as a more profound evaluation of specific parameters.

The evaluations showed that a five-phase model is able to depict customized teaching scenarios in a suitable way. The application is able to perform a variety of different moderate to complex scenarios in lectures with up to seventy students. But when performing very complex scenarios with a large number of rules and more than one hundred students, serious performance issues were observed.

When analyzing the parameters of ARSs in detail, it was discovered that students pay much attention to the technical performance and the overall overview about the current task. This overview includes the students' aversion to scroll during a task and the application's general color scheme. Students want to immediately recognize significant information and interaction elements on a screen. They do not like to scroll down to further answer possibilities, or up again to the current question and problem description. Using pictures and supporting elements (like progress bars) may be appreciated if these do not decrease the overall overview.

One could think that the students' acceptance of ARS tools is a short-term effect that vanishes as soon as ARSs become normal and therefore boring. But the novel *MobileQuiz2* offers the lecturers many possibilities to vary and differently use the tool, so that a "system fatigue" gets less probable. Nonetheless, the possibility to create customized teaching scenarios implies that lecturers have a fundamental knowledge in the field of instructional design. Furthermore, it requires a reasonable amount of creativity and audacity. In our evaluation, the lecturers stated that they wanted to have more predefined scenario blueprints to choose from. So, although the system offers various opportunities to improve the students' learning, using it in a didactically reasonable way faces the lecturers with noticeable challenges. Offering didactic support and qualification is essential for a successful application.

In addition to the implemented *MobileQuiz2* prototype, a graphical scenario editor was designed and implemented so support the lecturers in designing and customizing scenarios. The XML files, which are used to describe the teaching scenarios, required computer science skills, so that they could not easily be modified by an average lecturer outside of the field of computer sciences. The scenario editor had to be easy to use and flexible in order to not restrict the possibilities of the underlying model. The implementation and evaluation of the editor showed that even with graphical support, it is quite complex to create scenarios from scratch, as it needs a lot of abstract thinking and deep knowledge of the *MobileQuiz2*'s model.

An evaluation with four lecturers from different departments showed that the authoring tool was moderately rated in terms of ease of use, clear layout, and responsiveness. In this context it was mentioned that the representation of the scenario's structure appeared overloaded and confusing when looking at it for the first time. But most of the participants were still convinced and could imagine to continue using the tool in the future. Especially the availability of predefined scenarios was rated as very useful.

Therefore, further improvements are necessary to fully provide a scenario editor to lecturers. Especially a separation into normal usage and expert functionalities seems meaningful, as this would enable common lecturers to modify their scenarios in a controlled way and enable experts to create novel scenarios.

6.2 Outlook

With the start of the Fall semester 2016, the *MobileQuiz* is well accepted by lecturers and students of our university. It is firmly established at the University of Mannheim, and with the official release under the GNU General Public License, it is slowly adapted by other universities, schools and companies. The *MobileQuiz* will be further updated and improved in the future.

In addition, about twenty lecturers from different fields and universities were trained in the usage of the *MobileQuiz2*. The interest in the customizable quizzes increases beyond the University of Mannheim. Although the *MobileQuiz2* is only available as a prototype application, it is already used by lecturers and interested people of different fields and institutions, inside and outside of Germany. Therefore, our focus is now set on polishing the prototype for a smooth operation. This will include more functionalities and a better connection to the *MobileQuiz* and ILIAS.

In the medium term, the scenarios and quizzes should be interchangeable between the *MobileQuiz*, the *MobileQuiz2* and other e-learning applications of the university. In addition, it may be feasible to completely redesign the core of the *MobileQuiz2* prototype to take the performance issues into full consideration. With this, more specific learning analytics possibilities should be supported, and with the increasing number of active users, the effects of different novel scenarios on students should be evaluated more thoroughly.

Own Publications

- [1] Okan Aydingül, Philip Mildner, Stephan Kopf, Daniel Schön, and Wolfgang Effelsberg. EEOC - Easy Open Online Courses. A Platform to Create and Run Open Online Courses. In Saul Carliner, Catherine Fulford, and Nathaniel Ostashewski, editors, *Proceedings of EdMedia: World Conference on Educational Media and Technology 2015 of World Conference on Educational Media and Technology (EdMedia)*, pages 711–720, Montreal, Quebec, Canada, 2015. Association for the Advancement of Computing in Education (AACE).
- [2] Melanie Klinger, Daniel Schön, and Wolfgang Effelsberg. Implementierung und Evaluation eines QR-Code gestützten Umfrage-Tools für Präsenzveranstaltungen. In Andreas Breiter and Christoph Rensing, editors, *DeLFI 2013*, pages 119–130, Bremen, Germany, 2013. GI. ISBN 9783885796121.
- [3] Melanie Klinger, Marc Egloffstein, and Daniel Schön. Wir haben eine Lösung, aber wo ist das Problem? - Entwicklung von Entwurfsmustern für ARS-Szenarien auf Basis von empirischer Wirkungsforschung. In Hans Pongratz and Reinhard Keil, editors, *DeLFI 2015 - Die 13. e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V.*, pages 71–80, München, Germany, 2015. GI.
- [4] Daniel Schön, Melanie Klinger, Stephan Kopf, and Wolfgang Effelsberg. Mobile-Quiz—A Lecture Survey Tool using Smartphones and QR Tags. *International Journal of Digital Information and Wireless Communications (IJDIWC)*, 2(3):231–244, 2012.
- [5] Daniel Schön, Stephan Kopf, and Wolfgang Effelsberg. A Lightweight Mobile Quiz Application with Support for Multimedia Content. In *2012 International Conference on E-Learning and E-Technologies in Education (ICEEE)*, pages 134–139, Lodz, Poland, sep 2012. IEEE. ISBN 978-1-4673-1678-1. doi: 10.1109/ICeLeTE.2012.6333407.
- [6] Daniel Schön, Stephan Kopf, Stephan Schulz, and Wolfgang Effelsberg. Integrating a Lightweight Mobile Quiz on Mobile Devices into the Existing University Infrastructure. In *World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA) 2012*, pages 1901–1907, Denver, Colorado, USA, 2012. Association for the Advancement of Computing in Education (AACE).
- [7] Daniel Schön, Hendrik Lemelson, and Wolfgang Effelsberg. Situation-aware choice of the most accurate positioning system. In *2012 IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 259–264. IEEE, mar 2012. ISBN 978-1-4673-0907-3. doi: 10.1109/PerComW.2012.6197490.
- [8] Daniel Schön, Melanie Klinger, Stephan Kopf, and Wolfgang Effelsberg. HomeQuiz: Blending Paper Sheets with Mobile Self-Assessment Tests. In *World Conference on*

- Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA) 2013*, 2013.
- [9] Daniel Schön, Philip Mildner, Stephan Kopf, and Wolfgang Effelsberg. SMASH : Ein generisches System für interaktive Szenarien in der Vorlesung. In *DeLFI 2014 - Die 12. e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V.*, pages 294–296. GI, 2014.
 - [10] Daniel Schön, Steffen Sikora, Stephan Kopf, and Wolfgang Effelsberg. GLA: A Generic Analytics Tool for e-Learning. In Christoph Rensing and Stephan Trahasch, editors, *Proceedings of DeLFI Workshops 2014*, pages 112–115, Freiburg, Germany, 2014. CEUR.
 - [11] Daniel Schön, Melanie Klinger, Stephan Kopf, and Wolfgang Effelsberg. A Model for Customized In-Class Learning Scenarios An Approach to Enhance Audience Response Systems with customized Logic and Interactivity Benefits of Audience Response. In *Proceedings of the 7th International Conference on Computer Supported Education (CSEDU)*, pages 108–118, Lissabon, 2015. SCITEPRESS - Science and Technology Publications. ISBN 9789897581076. doi: 10.5220/0005449301080118.
 - [12] Daniel Schön, Licheng Yang, Melanie Klinger, Stephan Kopf, and Wolfgang Effelsberg. On the Effects of Different Parameters in Classroom Interactivity Systems on Students. In Saul Carliner, Catherine Fulford, and Nathaniel Ostashevski, editors, *Proceedings of EdMedia: World Conference on Educational Media and Technology 2015*, pages 559–567, Montreal, Quebec, Canada, 2015. Association for the Advancement of Computing in Education (AACE).
 - [13] Daniel Schön, Melanie Klinger, Stephan Kopf, and Wolfgang Effelsberg. Customized Teaching Scenarios for Smartphones in University Lecture Settings: Experiences with Several Teaching Scenarios Using the MobileQuiz2. In *Proceedings of the 8th International Conference on Computer Supported Education (CSEDU)*, Rom, Italy, 2016.
 - [14] Daniel Schön, Melanie Klinger, Stephan Kopf, Thilo Weigold, and Wolfgang Effelsberg. Customizable Learning Scenarios for Students’ Mobile Devices in Large University Lectures: A Next Generation Audience Response System. In Susan Zvacek, Teresa Maria Restivo, James Uhomoibhi, and Markus Helfert, editors, *Computer Supported Education: 7th International Conference, CSEDU 2015, Lisbon, Portugal, May 23-25, 2015, Revised Selected Papers*, pages 189–207. Springer International Publishing, 2016. ISBN 978-3-319-29585-5. doi: 10.1007/978-3-319-29585-5_11.

Supervised Student Theses

- [15] Nikolas Dürr. *Outsourcing of Expensive Computations From Mobile Devices*. Bachelor's thesis, University of Mannheim, 2012.
- [16] Ricardo Lürer. *Application for Synchronizing Questions (ASQ) - A Universal Interface for Generic Learning Contents*. Master's thesis, University of Mannheim, 2015.
- [17] Sadettin Öztürk. *Performance Messungen von AJAX Webanwendungen am Beispiel des MobileQuiz2*. Bachelor's thesis, University of Mannheim, 2015.
- [18] Steffen Sikora. *Learning Analytics - Guideline for the Usage in e-Learning Scenarios*. Bachelor's thesis, University of Mannheim, 2014.
- [19] Thilo Weigold. *Designing a Graphical Authoring Tool for Creating Classroom Learning Scenarios for Mobile Devices*. Bachelor's thesis, University of Mannheim, 2015.
- [20] Licheng Yang. *Parameter von Lehrszenarien mit mobilen Abstimmungssystemen in Universitären Vorlesungen und ihre Wirkung auf die Akzeptanz bei Studierenden und Lehrpersonen*. Bachelor's thesis, University of Mannheim, 2015.

Bibliography

- [21] Darren Abramson, Krzysztof Pietroszek, Leila Chinaei, Edward Lank, and Michael Terry. Classroom response systems in higher education: Meeting user needs with NetClick. In *2013 IEEE Global Engineering Education Conference (EDUCON)*, pages 840–846, Berlin, Germany, March 2013. IEEE. ISBN 978-1-4673-6110-1. doi: 10.1109/EduCon.2013.6530204.
- [22] Dimitra Adam, Dimitra Kioutsiouki, Anastasios Karakostas, and Stavros N. Demetriadis. Do Your Students Get It? Quiz It! The Android Classroom Response System. In *2014 IEEE 14th International Conference on Advanced Learning Technologies*, pages 168–170, Athens, Greek, July 2014. IEEE. ISBN 978-1-4799-4038-7. doi: 10.1109/ICALT.2014.57.
- [23] A. F. Agudo-Peregrina, A. Hernandez-Garcia, and . Iglesias-Pradas. Predicting academic performance with learning analytics in virtual learning environments: A comparative study of three interaction classifications. In *Computers in Education (SIIE), 2012 International Symposium on*, Andorra la Vella, 2012. IEEE. ISBN 978-1-4673-4743-3.
- [24] Stepeh M. Alessi and Stenley R.Trollip. *Learning Principles and Approaches*. Allyn & Bacon, Inc., 3rd edition, 2001. ISBN 9780205276912.
- [25] Stephen M. Alessi and Stanley R. Trollip. *Computer-based Instruction: Methods and Development*. Prentice-Hall, Inc., 1985. ISBN 0-13-164161-1.
- [26] Naif Radi Aljohani and Hugh C. Davis. Significance of Learning Analytics in Enhancing the Mobile and Pervasive Learning Environments. In *2012 Sixth International Conference on Next Generation Mobile Applications, Services and Technologies*, pages 70–74, Paris, September 2012. IEEE. ISBN 978-1-4673-2598-1. doi: 10.1109/NGMAST.2012.49.
- [27] Kimberly E. Arnold and Matthew D. Pistilli. Course signals at Purdue: using learning analytics to increase student success. In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge - LAK '12*, page 267, New York, New York, USA, 2012. ACM. ISBN 9781450311113. doi: 10.1145/2330601.2330666.
- [28] André F. S. Barbosa and Frutuoso G. M. Silva. Serious Games: design and development of OxyBlood. In *Proceedings of the 8th International Conference on Advances in Computer Entertainment Technology*, number June, pages 15:1–15:8, Lisbon, Portugal, 2011. ACM Press. ISBN 978-1-4503-0827-4. doi: 10.1145/2071423.2071442.

- [29] P. Baumgartner, H. Häfele, and K. Maier-Häfele. E-Learning Standards aus der didaktischen Perspektive. In *Campus 2002: Die virtuelle Hochschule in der Konsolidierungsphase*, pages 277–286. Waxmann, Münster, 2002.
- [30] Julija Lapuh Bele and Joze Rugelj. Comparing Learning Results of Web Based and Traditional Learning Students. In *Advances in Web-Based Learning-Icwl 2010*, volume 6483, pages 375–380. Springer Berlin Heidelberg, 2010. ISBN 978-3-642-17406-3. doi: 10.1007/978-3-642-17407-0_39.
- [31] F. Bellotti, R. Berta, A. de Gloria, and L. Primavera. Enhancing the educational value of video games. *Computers in Entertainment*, 7(2):1, June 2009. ISSN 15443574. doi: 10.1145/1541895.1541903.
- [32] Bhagyavati, Stan Kurkovsky, and Christopher C. Whitehead. Using asynchronous discussions to enhance student participation in CS courses. *ACM SIGCSE Bulletin*, 37(1):111, February 2005. ISSN 00978418. doi: 10.1145/1047124.1047393.
- [33] John Biggs. *Teaching for Quality Learning at University*. Maidenhead: The society for Research into Higher Education & Open Press University, 2003.
- [34] Donald Bligh. What’s the use of lectures? *Journal of Geography in Higher Education*, 9(1):105–106, April 2000. ISSN 0309-8265. doi: 10.1080/03098268508708932.
- [35] Anna Bouch, Allan Kuchinsky, and Nina Bhatti. Quality is in the eye of the beholder. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '00*, volume 2, pages 297–304, New York, New York, USA, 2000. ACM Press. ISBN 1581132166. doi: 10.1145/332040.332447.
- [36] Ann L. Brown. The advancement of learning. *Educational Researcher*, 23(8):4–12, 1994.
- [37] John Seely Brown, Allan Collins, and Paul Duguid. Situated Cognition and the Culture of Learning. *Educational Researcher*, 18(1):32–42, January 1989. ISSN 0013-189X. doi: 10.3102/0013189X018001032.
- [38] John T. Bruer. *Schools for Thought: A Science of Learning in the Classroom*. MIT Press, 1993. ISBN 9780262521963.
- [39] Jerome Bruner. *Toward a Theory of Instruction*. Belknap Press, 1974. ISBN 9780674897014.
- [40] Ralph A. Burns. Information Impact and Factors Affecting Recall. Technical report, Annual National Conference on Teaching Excellence and Conference of Administrators, Austin, Texas, 1985.
- [41] Ray A. Burnstein and Leon M. Lederman. Using wireless keypads in lecture classes. *The Physics Teacher*, 39(1):8, 2001. ISSN 0031921X. doi: 10.1119/1.1343420.
- [42] Jane E. Caldwell. Clickers in the Large Classroom: Current Research and Best-Practice Tips. *Cell Biology Education*, 6(1):9–20, March 2007. ISSN 1931-7913. doi: 10.1187/cbe.06-12-0205.
- [43] Nicola Capuano, Anna Pierri, Francesco Colace, Matteo Gaeta, and Giuseppina Rita Mangione. A mash-up authoring tool for e-learning based on pedagogical templates. In *Proceedings of the first ACM international workshop on Multimedia*

- technologies for distance learning - MTDL '09*, page 87, New York, New York, USA, 2009. ACM Press. ISBN 9781605587578. doi: 10.1145/1631111.1631126.
- [44] Sean Casey, Ben Kirman, and Duncan Rowland. The gopher game: a social, mobile, locative game with user generated content and peer review. In *Proceedings of the international conference on Advances in computer entertainment technology - ACE '07*, pages 9–16, Salzburg, Austria, 2007. ACM Press. ISBN 9781595936400. doi: 10.1145/1255047.1255050.
- [45] Sergio Luis Castillo and Gerardo Ayala. *Encyclopedia of the Sciences of Learning*. Number 1988. Springer US, Boston, MA, 2012. ISBN 978-1-4419-1427-9. doi: 10.1007/978-1-4419-1428-6.
- [46] M. Chabi and S. Ibrahim. The Impact of Proper Use of Learning System on Students' Performance - Case Study of Using MyMathLab. In *Proceedings of the 6th International Conference on Computer Supported Education*, pages 551–554, Barcelona, Spain, 2014. SCITEPRESS - Science and Technology Publications. ISBN 978-989-758-020-8. doi: 10.5220/0004961605510554.
- [47] Amanda Chaffin and Tiffany Barnes. Lessons from a course on serious games research and prototyping. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games - FDG '10*, pages 32–39, Monterey, California, 2010. ACM Press. ISBN 9781605589374. doi: <http://dx.doi.org/10.1145/1822348.1822353>.
- [48] Ching-Chiu Chao. An Investigation of Learning Style Differences and Attitudes toward Digital Game-based Learning among Mobile Users. In *2006 Fourth IEEE International Workshop on Wireless, Mobile and Ubiquitous Technology in Education (WMTE'06)*, pages 29–31. IEEE, November 2006. ISBN 0-7695-2723-X. doi: 10.1109/WMTE.2006.261340.
- [49] D. Charsky. From Edutainment to Serious Games: A Change in the Use of Game Characteristics. *Games and Culture*, 5(2):177–198, February 2010. ISSN 1555-4120. doi: 10.1177/1555412009354727.
- [50] Jenova Chen. Flow in games (and everything else). *Communications of the ACM*, 50(4):31, April 2007. ISSN 00010782. doi: 10.1145/1232743.1232769.
- [51] John C. Chen, Dexter C. Whittinghill, and Jennifer A. Kadlowec. Classes That Click: Fast, Rich Feedback to Enhance Student Learning and Satisfaction. *Journal of Engineering Education*, 99(2):159–168, April 2010. ISSN 10694730. doi: 10.1002/j.2168-9830.2010.tb01052.x.
- [52] Jui-Hung Chen, Timothy K. Shih, and Jui-Yi Chen. To develop the ubiquitous adventure RPG (role play game) game-based learning system. In *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2973–2978. IEEE, October 2012. ISBN 978-1-4673-1714-6. doi: 10.1109/ICSMC.2012.6378247.
- [53] Yuh-Ming Cheng and Peng-Fei Chen. Building an Online Game-Based Learning System for Elementary School. *2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, (59):35–38, 2008. doi: 10.1109/IIH-MSP.2008.328.

- [54] Michael G. Christel, Scott M. Stevens, Bryan S. Maher, Sean Brice, Matt Champer, Luke Jayapalan, Qiaosi Chen, Jing Jin, Daniel Hausmann, Nora Bastida, Xun Zhang, Vincent Alevan, Kenneth Koedinger, Catherine Chase, Erik Harpstead, and Derek Lomas. RumbleBlocks: Teaching science concepts to young children through a unity game. In *Proceedings of CGAMES'2012 USA - 17th International Conference on Computer Games: AI, Animation, Mobile, Interactive Multimedia, Educational and Serious Games*, pages 162–166, 2012. ISBN 9781467311199. doi: 10.1109/CGames.2012.6314570.
- [55] Michael G. Christel, Jessica Trybus, Siddhant D. Shah, Bo Hsuan Chang, Roma Dave, Adarshkumar Pavani, Ojas D. Sawant, Jimin Song, James Inglis, Sai Shashank Kairamkonda, Christian Karrs, Xuyan Ke, Eric Kron, and Xinghu Lu. Bringing Biome Exploration into the Classroom Through Interactive Tablet Experiences. *Lecture Notes in Computer Science*, 9090(May):91–102, 2015. doi: 10.1007/978-3-319-19126-3_8.
- [56] Andy Clark. *Being There: Putting Brain, Body, and World Together Again*. A Bradford Book, 1996. ISBN 9780262032407.
- [57] Doug Clow. The Learning Analytics Cycle: Closing the Loop Effectively. In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge - LAK '12*, pages 134–138, Vancouver, British Columbia, Canada, 2012. ACM. ISBN 9781450311113. doi: 10.1145/2330601.2330636.
- [58] Christina Conati and Xiaohong Zhao. Building and evaluating an intelligent pedagogical agent to improve the effectiveness of an educational game. In *Proceedings of the 9th international Conference on Intelligent User Interfaces*, pages 6–13, Funchal, Madeira, Portugal, 2004. ACM Press. ISBN 1581138156. doi: 10.1145/964442.964446.
- [59] Joseph Rene Corbeil and Maria Elena Valdes-Corbeil. Are you ready for mobile learning? *Educause Quarterly*, (2):51–58, 2007.
- [60] Jill E. Courte. Comparing student acceptance and performance of online activities to classroom activities. In *Proceeding of the 8th ACM SIG-information conference on Information technology education - SIGITE '07*, page 185, New York, New York, USA, 2007. ACM Press. ISBN 9781595939203. doi: 10.1145/1324302.1324341.
- [61] Ben Cowley, Darryl Charles, Michaela Black, and Ray Hickey. Toward an understanding of flow in video games. *Computers in Entertainment*, 6(2):1, July 2008. ISSN 15443574. doi: 10.1145/1371216.1371223.
- [62] D. R. Cruickshank and Ross Telfer. Classroom Games and Simulations. *Theory Into Practice*, 19(1):75–80, 1980.
- [63] Mihaly Csikszentmihalyi. *Flow : the psychology of optimal experience*. Harper-Perennial, New York, New York, USA, 2008. ISBN ISBN 978-0-06-016253-5.
- [64] Mihaly Csikszentmihalyi and Judith LeFevre. Optimal experience in work and leisure. *Journal of Personality and Social Psychology*, 56(5):815–822, 1989. ISSN 0022-3514. doi: 10.1037/0022-3514.56.5.815.

- [65] Nada Dabbagh, Angela D. Benson, André Denham, Roberto Joseph, Maha Al-Freih, Ghania Zgheib, Helen Fake, and Zhetao Guo. Mobile Learning. In *Learning Technologies and Globalization*, number Traxler 2007 in SpringerBriefs in Educational Communications and Technology, pages 15–19. Springer International Publishing, Cham, 2016. ISBN 978-3-319-22962-1. doi: 10.1007/978-3-319-22963-8_3.
- [66] Fred D. Davis. *A Technology Acceptance Model for Empirically Testing New End-User Information Systems: Theory and Results*. Ph. d. thesis, Massachusetts Institute of Technology, 1985.
- [67] Peter Dawabi, Laura Dietz, Alejandro Fernandez, and Martin Wessner. Concert-Studeo: Using PDAs to support face-to-face learning. In B Wasson, R Baggetun, U Hoppe, and S Ludvigsen, editors, *International Conference on Computer Support for Collaborative Learning 2003 - Community Events*, pages 235–237, Bergen, Norway, 2003.
- [68] Funda Dağ, Levent Durdu, and Serpil Gerdan. Evaluation of Educational Authoring Tools for Teachers Stressing of Perceived Usability Features. *Procedia - Social and Behavioral Sciences*, 116:888–901, February 2014. ISSN 18770428. doi: 10.1016/j.sbspro.2014.01.316.
- [69] Edward L. Deci and Richard M. Ryan. The Paradox of Achievement: The Harder You Push, the Worse it Gets. In *Improving Academic Achievement*, pages 61–87. Elsevier, 2002. doi: 10.1016/B978-012064455-1/50007-5.
- [70] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. From game design elements to gamefulness: Defining “Gamification”. In *Proceedings of the 15th International Academic MindTrek Conference on Envisioning Future Media Environments - MindTrek '11*, page 9, Tampere, Finland, 2011. ACM Press. ISBN 9781450308168. doi: 10.1145/2181037.2181040.
- [71] Michele D. Dickey. Engaging by design: How engagement strategies in popular computer and video games can inform instructional design. *Educational Technology Research and Development*, 53(2):67–83, 2005. ISSN 1042-1629. doi: 10.1007/BF02504866.
- [72] Andrea DiSessa. *Changing Minds Computers, Learning, and Literacy*. MIT Press, 2000. ISBN 9780262041805.
- [73] Damien Djaouti, Julian Alvarez, and Jean-Pierre Jessel. Can “Gaming 2.0” help design “Serious Games”? In *Proceedings of the 5th ACM SIGGRAPH Symposium on Video Games - Sandbox '10*, volume 1, pages 11–18, Los Angeles, California, USA, 2010. ACM Press. ISBN 9781450300971. doi: 10.1145/1836135.1836137.
- [74] Lars Doucet and Vinod Srinivasan. Designing entertaining educational games using procedural rhetoric. In *Proceedings of the 5th ACM SIGGRAPH Symposium on Video Games - Sandbox '10*, volume 1, pages 5–10, Los Angeles, California, 2010. ACM Press. ISBN 9781450300971. doi: 10.1145/1836135.1836136.
- [75] Hendrik Drachsler and Wolfgang Greller. The pulse of learning analytics understandings and expectations from the stakeholders. In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge - LAK '12*, pages 120–129, Vancouver, British Columbia, Canada, 2012. ACM. ISBN 9781450311113. doi: 10.1145/2330601.2330634.

- [76] Steve W. Draper, J. Cargill, and Q. Cutts. Electronically enhanced classroom interaction. *Australian Journal of Educational Technology*, 18(1):13–23, 2002. ISSN 0814-673X. doi: 10.14742/ajet.v18i1.1744.
- [77] Robert J. Dufresne, William J. Gerace, William J. Leonard, Jose P. Mestre, and Laura Wenk. Classtalk: A classroom communication system for active learning. *Journal of Computing in Higher Education*, 7:3–47, 1996. ISSN 10421726. doi: 10.1007/BF02948592.
- [78] A. L. Dyckhoff, Dennis Zielke, and Mareike Bültmann. Design and Implementation of a Learning Analytics Toolkit for Teachers. *Journal of Educational Technology & Society*, 15:58–76, 2012.
- [79] A. L. Dyckhoff, V. Lukarov, A. Muslim, M. A. Chatti, and U. Schroeder. Supporting action research with learning analytics. In *Proceedings of the Third International Conference on Learning Analytics and Knowledge - LAK '13*, pages 220–229, Leuven, Belgium, 2013. ACM. ISBN 9781450317856. doi: 10.1145/2460296.2460340.
- [80] Martin Ebner and Mandy Schiefner. Digital native students? Web 2.0-Nutzung von Studierenden. Technical report, TU Graz, Universität Zürich, 2009.
- [81] Mohamed E. Edrees. eLearning 2.0: Learning Management Systems Readiness. In *2013 Fourth International Conference on e-Learning "Best Practices in Management, Design and Development of e-Courses: Standards of Excellence and Creativity"*, pages 90–96. IEEE, may 2013. ISBN 978-0-7695-5036-7. doi: 10.1109/ECONF.2013.57.
- [82] Jan P. Ehlers, Daniel Möbs, J. vor dem Esche, K. Blume, H. Bollwein, and M. Halle. Einsatz von formativen, elektronischen Testsystemen in der Präsenzlehre. *GMS Zeitschrift für Medizinische Ausbildung*, 27, 2010. doi: 10.3205/zma000696.
- [83] Margarita Elkina, Alt Friedrichsfelde, Albrecht Fortenbacher, and Agathe Merceron. The learning analytics application LeMo – rationals and first results. *International Journal of Computing*, 12(3):290–293, 2013.
- [84] Mingyu Feng and Neil Heffernan. Can We Get Better Assessment from a Tutoring System Compared to Traditional Paper Testing? Can We Have Our Cake (Better Assessment) and Eat It too (Student Learning during the Test)? In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 6095 LNCS, pages 309–311. Springer Berlin Heidelberg, 2010. ISBN 364213436X. doi: 10.1007/978-3-642-13437-1_54.
- [85] Rebecca Ferguson. Learning analytics: drivers, developments and challenges. *International Journal of Technology Enhanced Learning*, 4(5/6):304, 2012. ISSN 1753-5255. doi: 10.1504/IJTEL.2012.051816.
- [86] Rebecca Ferguson and Simon Buckingham Shum. Social learning analytics. In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge - LAK '12*, page 23, Vancouver, British Columbia, Canada, 2012. ACM. ISBN 9781450311113. doi: 10.1145/2330601.2330616.

- [87] Carmen Fies and Jill Marshall. Classroom Response Systems: A Review of the Literature. *Journal of Science Education and Technology*, 15(1):101–109, March 2006. ISSN 1059-0145. doi: 10.1007/s10956-006-0360-1.
- [88] Zachary Fitz-Walter, Dian Tjondronegoro, and Peta Wyeth. Orientation Passport: using gamification to engage university students. *Proceedings of the 23rd Australian Computer-Human Interaction Conference on - OzCHI '11*, pages 122–125, 2011. doi: 10.1145/2071536.2071554.
- [89] Albrecht Fortenbacher, Marcus Klüsener, and Sebastian Schwarzrock. Ein generisches Datenmodell für Learning Analytics. *CEUR Workshop Proceedings*, pages 80–87, 2014.
- [90] Giovanni Fulantelli, Davide Taibi, and Marco Arrigo. A semantic approach to mobile learning analytics. In *Proceedings of the First International Conference on Technological Ecosystem for Enhancing Multiculturality - TEEM '13*, pages 287–292, Salamanca, Spain, 2013. ACM. ISBN 9781450323451. doi: 10.1145/2536536.2536579.
- [91] Daniel Gallego, Enrique Barra, Aldo Gordillo, and Gabriel Huecas. Enhanced recommendations for e-Learning authoring tools based on a proactive context-aware recommender. In *2013 IEEE Frontiers in Education Conference (FIE)*, pages 1393–1395, Oklahoma City, OK, USA, October 2013. IEEE. ISBN 978-1-4673-5261-1. doi: 10.1109/FIE.2013.6685060.
- [92] Gerald C Gannod, Janet E Burge, and Michael T Helmick. Using the inverted classroom to teach software engineering. In *Proceedings of the 13th international conference on Software engineering - ICSE '08*, page 777, New York, New York, USA, 2008. ACM Press. ISBN 9781605580791. doi: 10.1145/1368088.1368198.
- [93] Francisco J. García, Adriana J Berlanga, Maria N Moreno, Javier García, and Jorge Carabias. HyCo – An Authoring Tool to Create Semantic Learning Objects for Web-Based E-learning Systems. In *Web Engineering, Proceedings*, volume 3140, pages 344–348. Springer Berlin Heidelberg, 2004. ISBN 3-540-22511-0. doi: 10.1007/978-3-540-27834-4_42.
- [94] James Paul Gee. What video games have to teach us about learning and literacy. *Computers in Entertainment*, 1(1):20, October 2003. ISSN 15443574. doi: 10.1145/950566.950595.
- [95] Edward F. Gehringer and Barry W. Peddycord. The inverted-lecture model: a case study in computer architecture. In *Proceeding of the 44th ACM technical symposium on Computer science education - SIGCSE '13*, page 489, New York, New York, USA, 2013. ACM Press. ISBN 9781450318686. doi: 10.1145/2445196.2445343.
- [96] Michail N. Giannakos and Panayiotis Vlamos. Comparing a well designed webcast with traditional learning. In *Proceedings of the 2010 ACM conference on Information technology education - SIGITE '10*, page 65, New York, New York, USA, 2010. ACM Press. ISBN 9781450303439. doi: 10.1145/1867651.1867669.
- [97] Aldo Gordillo, Enrique Barra, and Juan Quemada. Enhancing web-based learning resources with quizzes through an authoring tool and an audience response system. In *2014 IEEE Frontiers in Education Conference (FIE) Proceedings*, pages 1–8,

- Madrid, Spain, October 2014. IEEE. ISBN 978-1-4799-3922-0. doi: 10.1109/FIE.2014.7044188.
- [98] Mark Green. Towards virtual environment authoring tools for content developers. In *Proceedings of the ACM symposium on Virtual reality software and technology - VRST '03*, page 117, New York, New York, USA, 2003. ACM Press. ISBN 1581135696. doi: 10.1145/1008653.1008675.
- [99] Wolfgang Greller and Hendrik Drachsler. Translating Learning into Numbers: A Generic Framework for Learning Analytics. *Educational Technology & Society*, 15(3):42–57, 2012.
- [100] Shirley Grundy. A Computer Adventure as a Worthwhile Educational Experience. *Interchange*, 22(4):41–55, 1991.
- [101] Lasse Hakulinen. Using serious games in computer science education. In *Proceedings of the 11th Koli Calling International Conference on Computing Education Research - Koli Calling '11*, pages 83–88, Koli, Finland, 2011. ACM Press. ISBN 9781450310529. doi: 10.1145/2094131.2094147.
- [102] Michael J. Hannafin. *The design, development & evaluation of instructional software*. Macmillan Publishing Co., Inc., 1988. ISBN 0-02-349990-7.
- [103] Johannes Harms, Dominik Seitz, Christoph Wimmer, Karin Kappel, and Thomas Grechenig. Low-Cost Gamification of Online Surveys : Improving the User Experience through Achievement Badges. *CHI PLAY'15 Proceedings of the 2nd ACM SIGCHI annual symposium on Computer-human interaction in play.*, pages 109–113, 2015. doi: 10.1145/2793107.2793146.
- [104] Alireza Hassanzadeh, Fatemeh Kanaani, and Shában Elahi. A model for measuring e-learning systems success in universities. *Expert Systems with Applications*, 39(12):10959–10966, September 2012. ISSN 09574174. doi: 10.1016/j.eswa.2012.03.028.
- [105] Mark Hendrikx, Sebastiaan Meijer, Joeri Van Der Velden, and Alexandru Iosup. Procedural content generation for games. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 9(1):1–22, February 2013. ISSN 15516857. doi: 10.1145/2422956.2422957.
- [106] Jim Hewitt. E-Learning. In Richard Gunstone, editor, *Encyclopedia of Science Education*, pages 362–363. Springer Netherlands, Dordrecht, 2015. ISBN 978-94-007-2149-4. doi: 10.1007/978-94-007-2150-0_36.
- [107] Adrian Holzer, Sten Govaerts, Andrii Vozniuk, Bruno Kocher, and Denis Gillet. Speakup in the classroom: Anonymous Temporary Social Media for Better Interactions Adrian. In *Proceedings of the extended abstracts of the 32nd annual ACM conference on Human factors in computing systems - CHI EA '14*, pages 1171–1176, Toronto, Ontario, Canada, 2014. ACM Press. ISBN 9781450324748. doi: 10.1145/2559206.2581211.
- [108] Peter C. Honebein. Seven goals for the design of constructivist learning environments. In *Constructivist Learning Environments: Case Studies in Instructional Design*, pages 17–24. Educational Technology Publications, New Jersey, 1996.

- [109] Gwo-Jen Hwang, Po-Han Wu, Ya-Yan Zhuang, Wan-Ling Kuo, and Yueh-Min Huang. An Investigation on Students' Cognitive Load and Learning Achievements for Participating in a Local Culture Mobile Learning Activity. In *2010 6th IEEE International Conference on Wireless, Mobile, and Ubiquitous Technologies in Education*, pages 27–33. IEEE, April 2010. ISBN 978-1-4244-6427-2. doi: 10.1109/WMUTE.2010.28.
- [110] L. Jackowska-Strumillo, J. Nowakowski, P. Strumillo, and P. Tomczak. Interactive question based learning methodology and clickers: Fundamentals of computer science course case study. In *2013 6th International Conference on Human System Interactions (HSI)*, pages 439–442. IEEE, June 2013. ISBN 978-1-4673-5637-4. doi: 10.1109/HSI.2013.6577862.
- [111] M. Jagar, J. Petrovic, and P. Pale. AuResS: The audience response system. In *ELMAR, 2012 Proceedings*, pages 171–174, Zadar, Croatia, September 2012.
- [112] Kwok Wai Jing, Neo Tse Kian, Neo Mai, Heidi Tan Yeen-Ju, Lai Chen Haw, and Son Yang Fook. Designing Opportunities to Learn through Online Active Learning Environment. In *2013 International Conference on Informatics and Creative Multimedia*, pages 93–98. IEEE, September 2013. ISBN 978-0-7695-5133-3. doi: 10.1109/ICICM.2013.23.
- [113] Kevin Johnson, Timothy Hall, and Derek O’Keeffe. Generation of Quiz Objects (QO) with a Quiz Engine Developer (QED). In *IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE’05)*, volume 2005, pages 120–122. IEEE, 2005. ISBN 0-7695-2385-4. doi: 10.1109/WMTE.2005.26.
- [114] David H. Jonassen. Computers as cognitive tools: Learning with technology, not from technology. *Journal of Computing in Higher Education*, 6(2):40–73, March 1995. ISSN 1042-1726. doi: 10.1007/BF02941038.
- [115] Eugene Judson. Learning from past and present: Electronic response systems in college lecture halls. *Journal of Computers in Mathematics and Science . . .*, 2002.
- [116] Robin H. Kay and Ann LeSage. Examining the Benefits and Challenges of Using Audience Response Systems: A Review of the Literature. *Comput. Educ.*, 53(3): 819–827, November 2009. ISSN 0360-1315. doi: 10.1016/j.compedu.2009.05.001.
- [117] David W. Keirse and Marilyn Bates. *Please Understand Me: Character and Temperament Types*. Prometheus Nemesis Book Company, Incorporated, 1984. ISBN 0960695400.
- [118] Caitlin Kelleher and Randy Pausch. Using storytelling to motivate programming. *Communications of the ACM*, 50(7):58, July 2007. ISSN 00010782. doi: 10.1145/1272516.1272540.
- [119] S. M. Keough. Clickers in the Classroom: A Review and a Replication. *Journal of Management Education*, 36(6):822–847, August 2012. ISSN 1052-5629. doi: 10.1177/1052562912454808.
- [120] Michael Kerres, Mark Bormann, and Marcel Vervenne. Didaktische Konzeption von Serious Games : Zur Verknüpfung von Spiel- und Lernangeboten. *Online*, (August):16, 2009. ISSN 1424-3636.

- [121] Ferial Khaddage and Fatmeh Zeidan. iPad in higher education: “Ready or not? a college case study in the middle east”. In *2012 15th International Conference on Interactive Collaborative Learning (ICL)*, pages 1–3. IEEE, September 2012. ISBN 978-1-4673-2427-4. doi: 10.1109/ICL.2012.6402161.
- [122] Firas Khatib, Frank DiMaio, Seth Cooper, Maciej Kazmierczyk, Mirosław Gilski, Szymon Krzywda, Helena Zabranska, Iva Pichova, James Thompson, Zoran Popović, Mariusz Jaskolski, and David Baker. Crystal structure of a monomeric retroviral protease solved by protein folding game players. *Nature Structural & Molecular Biology*, 18(10):1175–1177, September 2011. ISSN 1545-9993. doi: 10.1038/nsmb.2119.
- [123] Yeongjun Kim, Soonmook Jeong, Yongwoon Ji, Sangeun Lee, Key Ho Kwon, and Jae Wook Jeon. Smartphone Response System Using Twitter to Enable Effective Interaction and Improve Engagement in Large Classrooms. *IEEE Transactions on Education*, 58(2):98–103, may 2015. ISSN 0018-9359. doi: 10.1109/TE.2014.2329651.
- [124] Carlos Delgado Kloos, Abelardo Pardo, Pedro J. Munoz-Merino, Israel Gutierrez, and Derick Leony. Learning analytics @ UC3M. In *2013 IEEE Global Engineering Education Conference (EDUCON)*, pages 1232–1238, Berlin, March 2013. IEEE. ISBN 978-1-4673-6110-1. doi: 10.1109/EduCon.2013.6530264.
- [125] Stephan Kopf and Wolfgang Effelsberg. New Teaching and Learning Technologies for Interactive Lectures. *Advanced Technology for Learning*, 4(2), 2007. ISSN 1710-2251. doi: 10.2316/Journal.208.2007.2.208-1082.
- [126] Stephan Kopf, Nicolai Scheele, and Wolfgang Effelsberg. The Interactive Lecture: Teaching and learning technologies for large classrooms. *Department for Mathematics and Computer Science, University of Mannheim, Mannheim, GDR*, pages 1–18, 2005.
- [127] Steve Krug. *Don’t make me think*. New Riders, 2000. ISBN 789723107.
- [128] R. Kugler, P. Kunz, and U. Schütz. Einsatz von iPads im Studium. *Zeitschrift zu Theorie und Praxis der Aus- und Weiterbildung von Lehrerinnen und Lehrern*, 1: 1–31, 2013.
- [129] Dennis Kundisch, Philipp Herrmann, Michael Whittaker, Marc Beutner, Gregor Fels, Johannes Magenheimer, Michael Sievers, and Andrea Zoyke. Designing a web-based application to support Peer Instruction for very large Groups. In *Proceedings of the International Conference on Information Systems*, pages 1–12, Orlando, USA, 2012. AIS Electronic Library.
- [130] Dennis Kundisch, Johannes Magenheimer, Marc Beutner, Philipp Herrmann, Wolfgang Reinhardt, and Andrea Zokye. Classroom Response Systems. *Informatik-Spektrum*, 36(4):389–393, June 2013. ISSN 0170-6012. doi: 10.1007/s00287-013-0713-0.
- [131] Maureen J. Lage, Glenn J Platt, and Michael Treglia. Inverting the Classroom: A Gateway to Creating an Inclusive Learning Environment. *The Journal of Economic Education*, 31(1):30, January 2000. ISSN 00220485. doi: 10.2307/1183338.

- [132] Julija Lapuh Bele and Joze Rugelj. Comparing Efficiency of Web Based Learning Contents on Different Media. *International Journal of Emerging Technologies in Learning (iJET)*, 4(s3):31–36, 2009. ISSN 1863-0383. doi: 10.3991/ijet.v4s3.1087.
- [133] David L. Largent. Flipping a large CS0 course: an experience report about exploring the use of video, clickers and active learning. *Journal of Computing Sciences in Colleges*, 29(1):84–91, 2013.
- [134] Jean Lave. Teaching, as Learning, in Practice. *Mind, Culture, and Activity*, 3(3): 149–164, July 1996. ISSN 1074-9039. doi: 10.1207/s15327884mca0303_2.
- [135] M. Llamas-Nistal, M. Caeiro-Rodriguez, and J. Gonzalez-Tato. Web-based Audience Response System using the educational platform called BeA. In *Computers in Education (SIIE), 2012 International Symposium on*, pages 1–6, Andorra la Vella, Andorra, 2012.
- [136] Vlatko Lukarov, Mohaemd Amine Chatti, Hendrik Thüs, and Fatemeh Salehian Kia. Data Models in Learning Analytics. *CEUR Workshop Proceedings*, pages 88–95, 2014.
- [137] Leah P. Macfadyen and Shane Dawson. Numbers Are Not Enough. Why e-Learning Analytics Failed to Inform an Institutional Strategic Plan. *Journal of Educational Technology & Society*, 15(3):149–163, 2012. ISSN 14364522.
- [138] Lancelot A. MacManaway. Teaching methods in higher education-innovation and research. *Higher Education Quarterly*, 24(3):321–329, June 1970. ISSN 0951-5224. doi: 10.1111/j.1468-2273.1970.tb00346.x.
- [139] Andrew Macvean, Sanjeet Hajarnis, Brandon Headrick, Azriel Ferguson, Chinmay Barve, Devika Karnik, and Mark O Riedl. WeQuest: Scalable Alternate Reality Games Through End-User Content Authoring. In *Proceedings of the 8th International Conference on Advances in Computer Entertainment Technology - ACE '11*, pages 22:1–22:8, Lisbon, Portugal, 2011. ACM Press. ISBN 9781450308274. doi: 10.1145/2071423.2071450.
- [140] Thomas W. Malone and Mark R. Lepper. Making learning fun: A taxonomy of intrinsic motivations for learning. *Aptitude learning and instruction*, 3(3):223–253, 1987. ISSN 00376337. doi: 10.1016/S0037-6337(09)70509-1.
- [141] Estefania Martin, Carlos Lazaro, and Isidoro Hernan-Losada. Active learning in Telecommunication Engineering: A case study. In *IEEE EDUCON 2010 Conference*, pages 1555–1562. IEEE, 2010. ISBN 978-1-4244-6568-2. doi: 10.1109/EDUCON.2010.5492343.
- [142] Ouiame Filali Marzouki, Mohammed Khalidi Idrissi, and Samir Bennani. Mobile Education. In *Proceedings of International Conference on Advances in Mobile Computing & Multimedia - MoMM '13*, pages 567–575, Vienna, Austria, 2013. ACM Press. ISBN 9781450321068. doi: 10.1145/2536853.2536862.
- [143] D. A. McConnell, D. N. Steer, and K. D. Owens. Assessment and Active Learning Strategies for Introductory Geology Courses. *Journal of Geoscience Education*, 51(2):205–216, 2003. ISSN 10899995.

- [144] Séamus McLoone, Simon O’Keeffe, Rudi Villing, and Connor Brennan. Evaluation of a Smartphone-based Student Response System for Providing High Quality Real-time Responses in a Distributed Classroom. In *25th IET Irish Signals & Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communities Technologies (ISSC 2014/CIICT 2014)*, pages 210–215. Institution of Engineering and Technology, 2014. ISBN 978-1-84919-924-7. doi: 10.1049/cp.2014.0687.
- [145] Florian Mehm. Authoring serious games. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games - FDG ’10*, pages 271–273, Montaray, Calafornia, USA, 2010. ACM Press. ISBN 9781605589374. doi: 10.1145/1822348.1822390.
- [146] Philip Mildner, Christopher Campbell, and Wolfgang Effelsberg. Word Domination: Bringing Together Fun and Education in an Authoring-Based 3D Shooter Game. In Stefan Göbel and Josef Wiemeyer, editors, *Lecture Notes in Computer Science*, volume 8395 of *Lecture Notes in Computer Science*, chapter 7, pages 59–70. Springer International Publishing, 2014. ISBN 978-3-319-05971-6. doi: 10.1007/978-3-319-05972-3.
- [147] Philip Mildner, Benjamin John, Alexander Moch, and Wolfgang Effelsberg. Creation of custom-made serious games with user-generated learning content. *2014 13th Annual Workshop on Network and Systems Support for Games*, pages 1–6, 2014. doi: 10.1109/NetGames.2014.7008959.
- [148] Mark Miley, James A Redmond, and Colm Moore. Traditional Learning Vs. e-Learning - Some Results from Training Call Centre Personnel. In *Proceedings of the Tenth International Conference on Enterprise Information Systems*, pages 299–307. SciTePress - Science and and Technology Publications, 2008. ISBN 978-989-8111-36-4. doi: 10.5220/0001721302990307.
- [149] A.W. Mitchell and N.K. Nyland. Learning Styles of Faculty Members and Senior Students in Coordinated Programs in Dietetics Compared by Area of Career Interest. *Journal of the American Dietetic Association*, 97(9):A42, September 1997. ISSN 00028223. doi: 10.1016/S0002-8223(97)00469-0.
- [150] Markus Montola, Timo Nummenmaa, Andrés Lucero, Marion Boberg, and Hannu Korhonen. Applying game achievement systems to enhance user experience in a photo sharing service. *Proceedings of the 13th International MindTrek Conference: Everyday Life in the Ubiquitous Era*, 4(46):94–97, 2009. doi: 10.1145/1621841.1621859.
- [151] Andrew Moshirnia and Maya Israel. The Educational Efficacy of Distinct Information Delivery Systems in Modified Video Games. *Journal of Interactive Learning Research*, 21(3):383–405, 2010.
- [152] Pedro J. Muñoz-Merino, Carlos Delgado Kloos, and Jesús Fernández Naranjo. Enabling interoperability for LMS educational services. *Computer Standards & Interfaces*, 31(2):484–498, February 2009. ISSN 09205489. doi: 10.1016/j.csi.2008.06.009.
- [153] Trevor Murphy. Success and failure of audience response systems in the classroom. *Proceedings of the 36th annual ACM SIGUCCS conference on User services conference - SIGUCCS ’08*, page 33, 2008. doi: 10.1145/1449956.1449969.

- [154] Trevor Murphy, Kathy Fletcher, and Amy Haston. Supporting clickers on campus and the faculty who use them. In *Proceedings of the 38th annual fall conference on SIGUCCS - SIGUCCS '10*, SIGUCCS '10, page 79, New York, New York, USA, 2010. ACM Press. ISBN 9781450300032. doi: 10.1145/1878335.1878356.
- [155] T. Murray, Stephen Blessing, and S. Ainsworth. *Authoring Tools for Advanced Technology Learning Environments*. Springer Netherlands, Dordrecht, 2003. ISBN 978-90-481-6499-8. doi: 10.1007/978-94-017-0819-7.
- [156] Tom Murray. Authoring Intelligent Tutoring Systems: An analysis of the state of the art. *International Journal on Artificial Intelligence in Education(IJAIED)*, 10:98–129, 1999. ISSN 10980121.
- [157] Tom Murray, Beverly Woolf, and David Marshall. Lessons Learned from Authoring for Inquiry Learning: A Tale of Authoring Tool Evolution. In *Intelligent Tutoring Systems*, pages 197–206. Springer Berlin Heidelberg, 2004. ISBN 3-540-22948-5. doi: 10.1007/978-3-540-30139-4_19.
- [158] Jeanne Nakamura and Mihaly Csikszentmihalyi. The Concept of Flow Optimal Experience and Its Role in Development. *Handbook of positive psychology*, pages 89–105, 2002.
- [159] Kjetil L. Nielsen, Gabrielle Hansen, and John B. Stav. Teaching with student response systems (SRS): teacher-centric aspects that can negatively affect students' experience of using SRS. *Research in Learning Technology*, 21(1063519):1–13, June 2013. ISSN 2156-7077. doi: 10.3402/rlt.v21i0.18989.
- [160] Katja Niemann, Maren Scheffel, and Martin Wolpers. An overview of usage data formats for recommendations in TEL. In *Proceedings of the 2nd Workshop on Recommender Systems in Technology Enhanced Learning 2012*, volume 896, pages 95–100, Saarbrücken, Germany,, 2012. CEUR Workshop Proceedings.
- [161] James Oigara and Jared Keengwe. Students' perceptions of clickers as an instructional tool to promote active learning. *Education and Information Technologies*, 18(1):15–28, March 2013. ISSN 1360-2357. doi: 10.1007/s10639-011-9173-9.
- [162] Yeonjeong Park. A Pedagogical Framework for Mobile Learning: Categorizing Educational Applications of Mobile Technologies into Four Types. *The International Review of Reseach in Open and Distributed Learning*, 12(2), 2011.
- [163] Louise E. Parker and Mark R. Lepper. Effects of fantasy contexts on children's learning and motivation: Making learning more fun. *Journal of Personality and Social Psychology*, 62(4):625–633, 1992. ISSN 0022-3514. doi: 10.1037/0022-3514.62.4.625.
- [164] Barbara Patterson, Judith Kilpatrick, and Eric Woebkenberg. Evidence for teaching practice: The impact of clickers in a large classroom environment. *Nurse Education Today*, 30(7):603–607, October 2010. ISSN 02606917. doi: 10.1016/j.nedt.2009.12.008.
- [165] Roy D. Pea and Heidy Maldonado. WILD for learning : Interacting through new computing devices anytime , anywhere. *The Cambridge Handbook of the Learning Sciences*, pages 852–885, 2006.

- [166] Nancy E. Perry, Karen O. VandeKamp, Louise K. Mercer, and Carla J. Nordby. Investigating Teacher-Student Interactions That Foster Self-Regulated Learning. *Educational Psychologist*, 37(1):5–15, March 2002. ISSN 0046-1520. doi: 10.1207/S15326985EP3701_2.
- [167] Dominik Petko. Unterrichten mit Computerspielen: Didaktische Potenziale und Ansätze für den gezielten Einsatz in Schule und Ausbildung. *Zeitschrift für Theorie und Praxis der Medienbildung*, 15(16):15, 2008.
- [168] Marc Prensky. *Digital Game-Based Learning*. McGraw-Hill, 2001. ISBN 0071363440, 9780071363440.
- [169] Michael Prince. Does Active Learning Work? A Review of the Research. *Journal of Engineering Education*, 93(3):223–231, July 2004. ISSN 10694730. doi: 10.1002/j.2168-9830.2004.tb00809.x.
- [170] D. Rajavel and Rajesh Kushalkar. Clicker: Device Independent Student Response System, Developed for Classroom and Remote Learning to Provide Instant Response and Feedback. In *2014 IEEE Sixth International Conference on Technology for Education*, pages 62–65, Clappana, Inida, December 2014. IEEE. ISBN 978-1-4799-6489-5. doi: 10.1109/T4E.2014.4.
- [171] Paul Ramsden. *Learning to teach in higher education*, volume 24. Routledge, New York, NY, USA, 1992. ISBN 0415064155. doi: 10.1080/03075079312331382498.
- [172] W. Rascher, A. Ackermann, and I. Knerr. Interaktive Kommunikationssysteme im kurrikularen Unterricht der Pädiatrie für Medizinstudierende. *Monatsschrift Kinderheilkunde*, 152:432–437, 2003.
- [173] N. W. Reay, Pengfei Li, and Lei Bao. Testing a new voting machine question methodology. *American Journal of Physics*, 76(2):171, 2008. ISSN 00029505. doi: 10.1119/1.2820392.
- [174] Sheryl Wetter Riechmann and Anthony F. Grasha. A Rational Approach to Developing and Assessing the Construct Validity of a Student Learning Style Scales Instrument. *The Journal of Psychology*, 87(2):213–223, July 1974. ISSN 0022-3980. doi: 10.1080/00223980.1974.9915693.
- [175] Michael K. Salemi. Clickenomics: Using a Classroom Response System to Increase Student Engagement in a Large-Enrollment Principles of Economics Course. *The Journal of Economic Education*, 40(4):385–404, October 2009. ISSN 0022-0485. doi: 10.1080/00220480903237950.
- [176] S. Sayago, J. Martínez, J. Blat, R. García, D. Griffiths, and F Casado. Design and evaluation of a simple eLearning authoring tool. In *HCI related papers of Interacción 2004*, pages 81–88. Kluwer Academic Publishers, Dordrecht, 2006. doi: 10.1007/1-4020-4205-1_7.
- [177] Nicolai Scheele, Anja Wessels, Wolfgang Effelsberg, Manfred Hofer, and Stefan Fries. Experiences with Interactive Lectures: Considerations from the Perspective of Educational Psychology and Computer Science. In *Proceedings of the 2005 Conference on Computer Support for Collaborative Learning: Learning 2005: The Next 10 Years!*, CSCL '05, pages 547–556. International Society of the Learning Sciences, 2005. ISBN 0-8058-5782-6.

- [178] Eusebio Scornavacca, Sid Huff, and Stephen Marshall. Mobile phones in the classroom. *Communications of the ACM*, 52(4):142, April 2009. ISSN 00010782. doi: 10.1145/1498765.1498803.
- [179] Edgar Seemann. Teaching Mathematics in Online Courses - An Interactive Feedback and Assessment Tool. In *Proceedings of the 6th International Conference on Computer Supported Education*, pages 415–420, Barcelona, Spain, 2014. SCITEPRESS - Science and Technology Publications. ISBN 978-989-758-020-8. doi: 10.5220/0004939204150420.
- [180] H.A. Sholl, R.A. Ammar, and W.S. Weiss. Using HyperCard to rapidly prototype human-computer interfaces to CASE systems. In *Conference Proceedings., IEEE International Conference on Systems, Man and Cybernetics*, pages 1266–1267, Cambridge, MA, USA, 1989. IEEE. doi: 10.1109/ICSMC.1989.71506.
- [181] Keng Siau, Hong Sheng, and Fiona Fui-Hoon Nah. Use of a Classroom Response System to Enhance Classroom Interactivity. *IEEE Transactions on Education*, 49(3):398–403, August 2006. ISSN 0018-9359. doi: 10.1109/TE.2006.879802.
- [182] George Siemens and Ryan S. J. d. Baker. Learning Analytics and Educational Data Mining: Towards Communication and Collaboration. In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge - LAK '12*, pages 252–254, Vancouver, British Columbia, Canada, 2012. ACM. ISBN 9781450311113. doi: 10.1145/2330601.2330661.
- [183] R. E. Slavin. *Cooperative learning. Theory research and practice*. Allyn and Bacon, Boston, 1995. ISBN 0-205-15630-4.
- [184] Øystein Sørebo, Hallgeir Halvari, Vebjørn Flaata Gulli, and Roar Kristiansen. The role of self-determination theory in explaining teachers' motivation to continue to use e-learning technology. *Computers & Education*, 53(4):1177–1187, December 2009. ISSN 03601315. doi: 10.1016/j.compedu.2009.06.001.
- [185] Marcus Specht. E-Learning Authoring Tools. In Norbert M. Seel, editor, *The Sciences*, pages pp 1111–1113. Springer US, Boston, MA, 2012. ISBN 978-1-4419-1427-9. doi: 10.1007/978-1-4419-1428-6.
- [186] Maddalena Taras. To Feedback or Not to Feedback in Student Self-assessment. *Assessment & Evaluation in Higher Education*, 28(5):549–565, October 2003. ISSN 0260-2938. doi: 10.1080/02602930301678.
- [187] Stephanie D. Teasley and Steven Lonn. Using Learning Management Systems to support students' collaborative learning in higher education. In *Proceedings of the 8th International Conference on Computer Supported Collaborative Learning*, volume 8, pages 718–720. International Society of the Learning Sciences, 2007. ISBN 9780615154374.
- [188] Scott Teel, Dino Schweitzer, and Steven Fulton. Braingame: A Web-based Student Response System. *J. Comput. Sci. Coll.*, 28(2):40–47, 2012. ISSN 1937-4771.
- [189] Anne-Maritta Tervakari, Jarno Marttila, Meri Kailanto, Jukka Huhtamäki, Juho Koro, and Kirsi Silius. Developing Learning Analytics for TUT Circle. In *IFIP Advances in Information and Communication Technology*, volume 395, pages 101–110. Springer Berlin Heidelberg, 2013. ISBN 9783642372841. doi: 10.1007/978-3-642-37285-8.11.

- [190] Javier Torrente, Ángel del Blanco, Guillermo Cañizal, Pablo Moreno-Ger, and Baltasar Fernández-Manjón. "e-Adventure3D": An Open Source Authoring Environment for 3D Adventure Games in Education. In *Proceedings of the 2008 International Conference in Advances on Computer Entertainment Technology - ACE '08*, number August, page 191, Yokohama, Japan, 2008. ISBN 9781605583938. doi: 10.1145/1501750.1501795.
- [191] John Traxler. Defining, Discussing and Evaluating Mobile Learning: The moving finger writes and having writ *The International Review of Research in Open and Distributed Learning*, 8, 2007.
- [192] Eric Tremblay. Educating the Mobile Generation – using personal cell phones as audience response systems in post-secondary science teaching. *Journal of Computers in Mathematics and Science Teaching*, 29:217–227, 2010. ISSN 0731-9258.
- [193] Chung Cheng Tseng, Chung Hsien Lan, and K. Robert Lai. Modeling Beer Game as Role-Play Collaborative Learning via Fuzzy Constraint-Directed Agent Negotiation. *2008 Eighth IEEE International Conference on Advanced Learning Technologies*, pages 634–638, 2008. doi: 10.1109/ICALT.2008.107.
- [194] Matti Uhari, Marjo Renko, and Hannu Soini. Experiences of using an interactive audience response system in lectures. *BMC medical education*, 3:12, 2003. ISSN 1472-6920. doi: 10.1186/1472-6920-3-12.
- [195] Judy Unrein. Rapid Power Tools: The top performers of eLearning authoring software. *eLearn*, 2012(5):5, may 2012. ISSN 1535394X. doi: 10.1145/2207270.2221186.
- [196] Kalyan Veeramachaneni and Franck Dernoncourt. Moocdb: Developing data standards for mooc data science. In *In Proceedings of the 1st Workshop on Massive open Online Courses at the 16th Annual Conference on Artificial Intelligence in Education*, pages 1–8, Memphis, 2013.
- [197] Robert Vinaja. The Use of Lecture Videos, eBooks, and Clickers in Computer Courses. *Journal of Computing Sciences in Colleges*, 30(2):23–32, 2014. ISSN 1937-4771.
- [198] Luis von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the 2004 conference on Human factors in computing systems - CHI '04*, volume 6, pages 319–326, New York, New York, USA, 2004. ACM Press. ISBN 1581137028. doi: 10.1145/985692.985733.
- [199] Luis von Ahn and Laura Dabbish. Designing games with a purpose. *Communications of the ACM*, 51(8):57, August 2008. ISSN 00010782. doi: 10.1145/1378704.1378719.
- [200] Luis von Ahn, Ruoran Liu, and Manuel Blum. Peekaboom: A Game for Locating Objects in Images. *Proceedings of the SIGCHI conference on Human Factors in computing systems - CHI '06*, page 55, 2006. doi: 10.1145/1124772.1124782.
- [201] Luis von von Ahn, Shiry Ginosar, Mihir Kedia, and Manuel Blum. Improving Image Search with PHETCH. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, pages IV–1209–IV–1212. IEEE, 2007. ISBN 1-4244-0727-3. doi: 10.1109/ICASSP.2007.367293.

- [202] Ellen Wagner and Phil Ice. Data Changes Everything: Delivering on the Promise of Learning Analytics in Higher Education. *Educause Review*, 47(4):32–36, 2012.
- [203] Günter Wallner and Simone Kriglstein. DOGeometry: Teaching geometry through play. In *Proceedings of the 4th International Conference on Fun and Games*, pages 11–18, Toulouse, France, 2012. ACM Press. ISBN 9781450315708. doi: 10.1145/2367616.2367618.
- [204] Yi-Shun Wang, Hsiu-Yuan Wang, and Daniel Y. Shee. Measuring e-learning systems success in an organizational context: Scale development and validation. *Computers in Human Behavior*, 23(4):1792–1808, July 2007. ISSN 07475632. doi: 10.1016/j.chb.2005.10.006.
- [205] Katrin Weber and Bernd Becker. Formative Evaluation des mobilen Classroom-Response-Systems SMILE. In *E-Learning zwischen Vision und Alltag: zum Stand der Dinge*, number 1985, pages 277–289, Frankfurt, 2013. Waxmann.
- [206] René Wegener, Andreas Prinz, and Jan Marco Leimeister. Entwicklung innovativer, mobiler Lernanwendungen für den Einsatz in Massenveranstaltungen. In *Konferenz Mobile und Ubiquitäre Informationssysteme (MMS), Kaiserslautern Germany*, number 2011, 2011. ISBN 978-3-88579-279-6.
- [207] René Wegener, Andreas Prinz, and Jan Marco Leimeister. Mobiles Lernen mit Tablet PCs in universitären Massenveranstaltungen – Eine empirische Untersuchung von Einflussfaktoren auf Nutzungsintention und Lernendenzufriedenheit. In *Mobile Learning*, pages 101–120. Springer Fachmedien Wiesbaden, Wiesbaden, 2013. ISBN 9781118894309. doi: 10.1007/978-3-531-19484-4_6.
- [208] Brent Wilson and Peggy Cole. A review of cognitive teaching models. *Educational Technology Research and Development*, 39(4):47–64, December 1991. ISSN 1042-1629. doi: 10.1007/BF02296571.
- [209] Yang Xiao. An analysis for differentiated services in IEEE 802.11 and IEEE 802.11e wireless LANs. In *24th International Conference on Distributed Computing Systems, 2004. Proceedings.*, pages 32–39. IEEE, 2004. ISBN 0-7695-2086-3. doi: 10.1109/ICDCS.2004.1281565.
- [210] Sugi Yuuichi, Kita Toshihiro, Yasunami Seisuke, and Nakano Hiroshi. Web-based Rapid Authoring Tool for LMS Quiz Creation. In *2006 7th International Conference on Information Technology Based Higher Education and Training*, pages 617–620. IEEE, July 2006. ISBN 1-4244-0405-3. doi: 10.1109/ITHET.2006.339676.
- [211] Raphael Zender, Peter Haucke, and Ulrike Lucke. Click2Vote – Systematische Integration heterogener Lösungen. In *DeLFI 2014 - Die 12. e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V.*, pages 163–168, Freiburg, 2014. Lecture Notes in Informatics.
- [212] Michael Zyda, John Hiles, Alex Mayberry, Casey Wardynski, Michael Capps, Brian Osborn, Russell Shilling, Martin Robaszewski, and Margaret Davis. Entertainment R&D for Defense. *IEEE Computer Graphics and Applications*, 23(1):28–36, 2003. ISSN 0272-1716. doi: 10.1109/mcg.2003.1159611.

Glossary

AJAX *Asynchronous JavaScript and XML* describes a concept of asynchronous data transfer between a browser and server. It enables the browser to send independent HTTP-requests to gather further information and change the current page visualization directly, without reloading the complete page. 5, 26, 71, 75, 157

AngularJS This client-side JavaScript framework is mainly designed for single-page web applications with a MVC structure. It is published under an open source license by Google Inc..
<https://angularjs.org/>. 83, 84

Apache HTTP Server Mostly only called Apache, it is one of the world's most used web server applications. It is developed and maintained by an open community under the *Apache Software Foundation*. It is released under the *Apache License* as a free and open-source software.
<https://httpd.apache.org/>. 63, 70

ARS An *Audience Response System* is a system which supports the interaction of an audience with the presenter. Such system mostly use wireless clickers to send the individual opinion of every listener to a central knot which summarizes the results. There are many known notations for them: Classroom Response Systems (CRS), Electronic Voting Systems (TED) or Clicker Systems. 1–3, 38, 44, 45, 68, 122, 127, 130, 131

COTS *Commercial of the shelf* describes regular computer games created for joy and fun. They are usually designed for leisure time but can sometimes also be used for serious purposes. 12, 15

CSS *Cascading Style Sheets* is a stylesheet-language for electronic documents and one of the core languages of the World Wide Web. It is mostly used to describe the layout and design of web pages. 64, 71, 75, 155

CSS3 *Cascading Style Sheets 3* is a stylesheet-language and together with HTML5 one of the core languages of the World Wide Web. *CSS3* is the newest version of CSS. 71

CSV *Comma Separated Value* describes the structure text files for storing or exchanging simple structured data. There is no common standard, but it is commonly used for tables and lists where the individual information is separated by a comma or semicolon
. 80

DOM *Document Object Model* describes a language-independent and cross-platform convention for representing and interacting with objects. Among others, it is used in HTML5 web documents. It organizes its nodes in a tree-like structure and offers an interface to address and manipulate this. 75, 84

ER Diagram An *Entity Relationship Diagram* is used to describe inter-related things in a specific domain of knowledge. It is composed of entity types and relationships between those. 65

Flash *Adobe Flash* is the name of a platform to program and execute multimedia data and interactive content. It is mostly used for interactive content within web pages. 26

GNU General Public License This license is one of the most used software licenses in the world. It allows to study, use, modify and distribute the original work. When software under this license is used within another product, the new product has to grant the same rights. 122, 132

GUID A *Globally Unique Identifier* is a globally unique id used for objects and database entries. They are e.g., used for identifying data volumes in operating systems. 66

Highcharts It is a pure JavaScript library developed by *Highsoft*, Norway. It can be used to plot charts and figures live on a website. *Highcharts* is free for non-commercial purposes.
<http://www.highcharts.com>. 64

HTML *Hypertext Markup Language* is the most common language to describe web pages. Web browsers read HTML along with CSS and JavaScript and render them into the visible view. 64, 67, 68, 84

HTML5 The fifth version of the *HyperText Markup Language* which is used to describe, visualize, and cross-link documents. It is commonly used in the World Wide Web. Version five offers new functionality for video, audio, local memory storage, 2D and 3D visualization. 26, 64, 67, 71, 130, 156

HTTP *Hypertext Transfer Protocol* is an application protocol for hypermedia information systems. It is the foundational protocol for the World Wide Web. 155

ILIAS The *Integrated Learning, Information, and Work Cooperation System* is an open source learning platform. It is the default learning management system of the University of Mannheim and many other universities in Germany. It is published under the terms of the GNU General Public License. <http://www.ilias.de>. 12, 40–42, 45, 61–65, 67, 68, 70, 80, 89, 122, 129, 132

IMS Global The *IMS Global Learning Consortium* is a global, nonprofit organization that strives to increase the impact and growth of learning technologies.
<https://www.imsglobal.org/>. 28

JavaScript A scripting language which was originally developed for dynamic web pages, It enables the browser to modify page content and layout without reloading the whole page. Nowadays, *JavaScript* can also be used in server applications. 5, 71, 72, 83

- jqPlot** As an open source charting and plotting plugin for the javascript framework jQuery, it can produce several chart types and offers a high variety of additional features. It was originally developed by Chris Leonello and is now maintained by Paul Pritchard. *jqPlot* is licensed under the MIT and GPL version 2 licenses.
<http://http://www.jqplot.com/>. 71, 72
- jQuery** It is one of the most popular cross platform JavaScript libraries today. It was designed to simplify client-side scripting. As a free, open-source software, it is published under the MIT License.
<https://jquery.com/>. 71, 93, 157
- jQuery Mobile** *jQuery Mobile* is JavaScript library based on JQuery. It is specialized for touch sensitive, mobile surfaces and focuses on the compatibility with a wide variety of smartphones and tablet computers.
<https://jquerymobile.com/>. 61, 64, 66, 71
- JSON** *JavaScript Object Notation* is a lightweight data-change format which strives to be better human readable than comparative formats like XML. It is commonly used in AJAX applications.
<http://www.json.org/>. 66
- Likert** *Likert-scales* are a widely used approach to scale responses in questionnaires. It is named after his inventor Rensis Likert. 77, 85, 88
- LOM** The *Learning Objects Metadata* is an open standard for metadata on learning objects passed by the Institute of Electrical and Electronics Engineers (IEEE). 26
- MIMO** *Multiple-input and multiple-output* describes a method in wireless communication for increasing the capacity of a radio link by multiplying the antennas of transmitters and receivers.. 94
- MIT License** The free software license was defined at the Massachusetts Institute of Technology (MIT). It puts very little restrictions on the reuse of the source code and there fore offers a very high compatibility. It permits the proprietary use of the software. 157
- MOOC** *Massive Open Online Courses* are free online courses with a huge amount of participants. They are usually held at university level and have a high usage of videos, reading materials, forums and communication features. 11, 29
- Moodle** A free and object-oriented software for course management and learning management which is written in PHP and uses the GNU GPLv3+ license.
<https://moodle.org>. 12
- MVC** *Model View Controller* describes a coding pattern which structures the program in three components. The Model files contain the data and the logic of the program. View classes describe the appearance and output whereas the controller classes control the navigation through the application. 67, 155
- MySQL** An open-source, relational database management system, it is published under the GNU General Public License as well as under several proprietary agreements. *MySQL* is a popular choice for web applications. It is developed by the Oracle Corporation.
<https://www.mysql.com/>. 63, 65

NTT DoCoMo The largest cellphone provider of Japan.

<https://www.nttdocomo.co.jp/english/>. 64

PDA *Personal Digital Assistants* are compact and mobile computers which are mostly used for calendar, address, and task management. They are almost completely replaced by smartphones nowadays. 1

PEAR *PHP Extension and Application Repository* is a framework and distribution system for PHP components.

<http://pear.php.net>. 65

PHP The recursive acronym for *PHP: Hypertext Preprocessor* describes a scripting language which is mainly used for dynamic web pages. It is one of the most common languages used in the web today.

<https://secure.php.net/>. 61, 63, 67, 68, 70, 71, 83, 159

QR code A *Quick Response Code* is a two dimensional, matrix bar code. It is an machine-readable optical label which is mostly used to encode numeric and alphanumeric information. Therefore it became very popular to encode the URLs to web pages. 4–6, 32, 40, 41, 43, 50, 58, 62–64, 68–71, 73–75, 78, 84, 91, 92, 101, 107, 111, 114, 125, 127, 129, 130

SCORM The *Sharable Content Object Reference Model* is a reference model for shareable electronic learning materials.. 26

Sequences A structure pattern used in databases for creating unique values for identifying objects and data. 65

Skyrim A role playing computer game, developed by Bethesda Game Studios and released on 11th November 2011. It is the fifth part in a row of computer games and plays in an area called Skyrim. 36

Stack Overflow An Internet platform where users can ask questions about software development. The answers are given by other users and the best answers can be rated by the community to show up on top of the page.

<http://stackoverflow.com>. 38

Twitter A real-time micro-blogging application. The communication is done with so called *tweets* which only allow 140 characters.

<https://twitter.com>. 79

URL A *Uniform Resource Locator* identifies and localizes a resource and the mechanism for retrieving in computer networks. They are most commonly used to reference web pages. 4, 32, 43, 50, 58, 62, 64, 68, 70, 73, 75, 92, 101, 107, 125, 129, 130

URL shortening service A service that returns a significant shorter URL to a given input URL. It is mostly used to provide readable, easily remembered strings to a long and technical URL. The service provider stores a reference from the shortened to the original URL and redirects the clients to their destination. 64, 92

Wikipedia The world's most famous free-access Internet encyclopedia.

<https://en.wikipedia.com>. 66

Word Domination A first-person shooter game, developed at the University of Mannheim. It features the embedding of knowledge quizzes into the gaming experience.
http://www.knowledge-gaming.de/games/word_domination/. 37

Youtube A global video-sharing website and is now owned by Google Inc. Users can upload own videos and watch videos uploaded by others.
<https://www.youtube.com>. 67

Zend The *Zend framework* is an open source, object-oriented framework for PHP web applications.
In this work, the framework was used in its version 2.
<http://framework.zend.com/>. 71, 80, 83

Zork One of the earliest interactive fiction computer games. The first version was written in 1977-1979. In comparison to modern computer games, the player controls his character with text input of full expressions, which are interpreted by a text parser which was quite sophisticated at the time of the first release. 36

Abbreviations

AJAX	A synchronous J ava S cript and X ml
ARS	A udience R esponse S ystem
COTS	C ommercial O f T he S helf
CRS	C lassroom R esponse S ystem
CSS	C ascading S tyle S heets
CSV	C omma S eparated V alues
DOM	D ocument O bject M odel
GUID	G lobally U nique I dentifier
HTML	H yper T ext M arkup L anguage
JSON	J ava S cript O bject N otation
LOM	L earning O bject M etadata
MIMO	M ultiple- I nter M ultiple- O utput
MVC	M odel V iew C ontroller
PDA	P ersonal D igital A ssistant
PEAR	P HP E xtension and A pplication R epository
PRS	P ersonal R esponse S ystem
SCORM	S harable C ontent O bject R eference M odel
SRS	S tudent R esponse S ystem
QR	Q uick R esponse

Appendix A

Diagrams

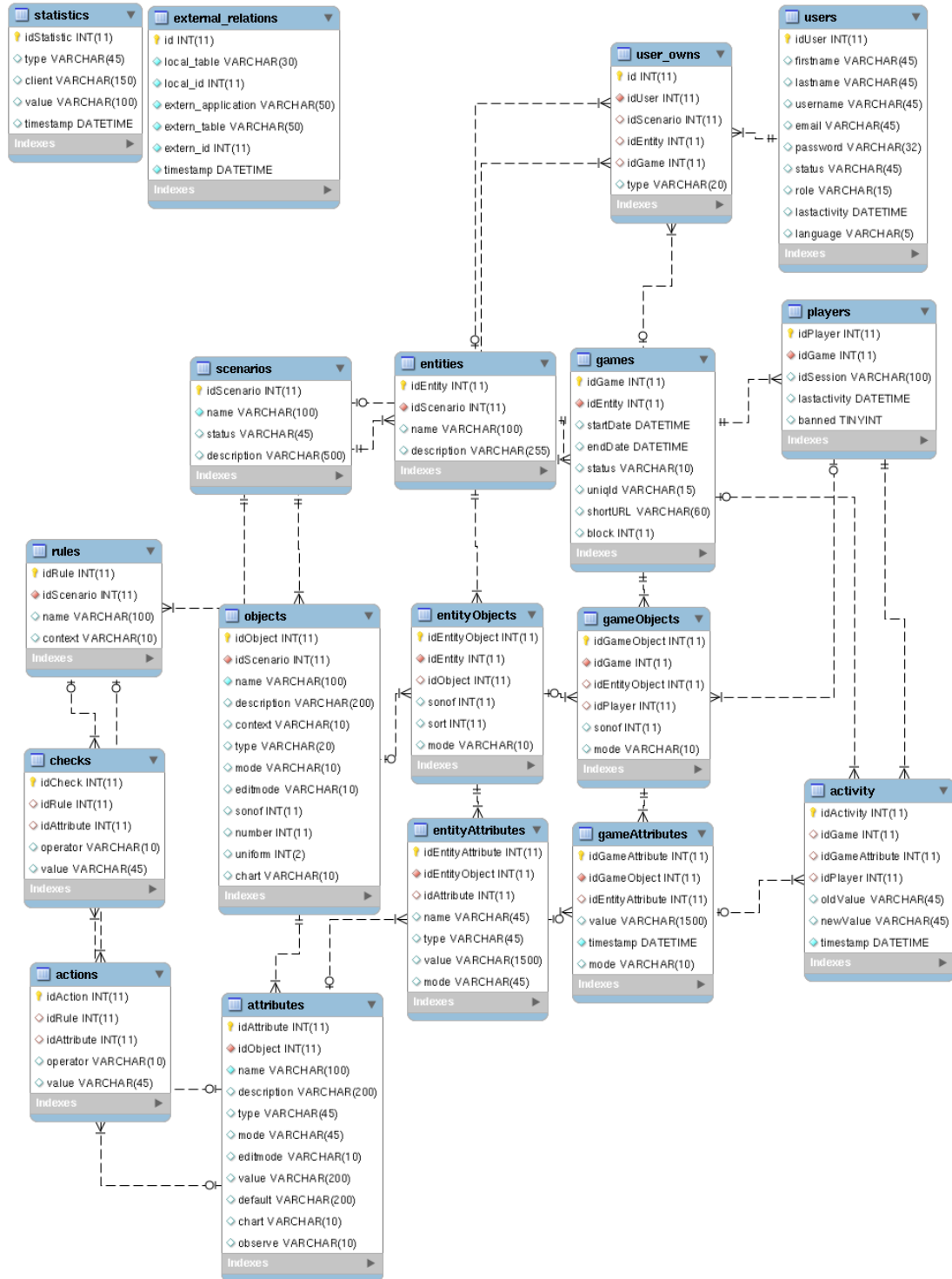


FIGURE A.1: ER diagram of the full database scheme of the *MobileQuiz2*.

Appendix B

Survey for the students about the *MobileQuiz* in 2012

QuizApp – Survey

I participated in the quiz...

☐ Yes, with...
 ☐ No, because..

☐ a smartphone
 ☐ I tried, but it did not work.

☐ a laptop
 ☐ It was too troublesome to participate.

☐ an ordinary mobile phone
 ☐ I did not bring a device with me, but own one.

☐ other:
 ☐ I do not own a smartphone, laptop or similar device.

☐ other:

If you participated please complete the following fields:

Your seat number	Model of your device	Megapixel of your camera

Via which network did you participate

☐ WiFi of the university (Adonis, Doniluma, ...)
 ☐ data net of my mobile network provider

The access to the quiz was...

☐ really fast
 ☐ fast
 ☐ ok
 ☐ slow
 ☐ way too slow

The technical handling of the quiz for me was...

☐ really easy
 ☐ easy
 ☐ ok
 ☐ difficult
 ☐ very difficult

Did the quiz make the lecture more interesting for you?

☐ yes
 ☐ a bit
 ☐ maybe
 ☐ no
 ☐ not at all

Did the quiz help you to understand the lecture content (e. g. by the possibility to test your understanding of the contents)?

☐ yes
 ☐ a bit
 ☐ maybe
 ☐ no
 ☐ not at all

Did the quiz increase your motivation to learn the lecture contents?

☐ yes
 ☐ a bit
 ☐ maybe
 ☐ no
 ☐ not at all

Do you want to have such quizzes more often during a lecture?

☐ yes
 ☐ maybe
 ☐ no

Comments:

All information is absolutely anonymous.

FIGURE B.1: Sheet for the survey about the *MobileQuiz* in 2012.

Participated in Quizzes	
yes	201
no	52

Participated with device:	
smartphone	175
laptop	8
ordinary phone	2
other	13

Reason for not participating:	
didn't work	8
too troublesome	2
not bring a device with me	32
do not own a device	5
other	4

Used devices:	
Blackberry	4
G1	1
Galaxy	1
HTC	25
Ipad	4
Iphone	70
Ipod	5
Laptop	3
LG	3
MacAir	1
MacBook	3
Motorola	6
Nexus	1
Samsung	44
Sony	9

Megapixels of camera:	
1,3	1
2	5
3	8
5	31
8	42
9	1
12	1

Via which network did you participate?	
Wifi of the university	156
data net of my mobile network provider	41

FIGURE B.2: Students' answers in the survey about the *MobileQuiz* in economics in 2012. (1/2)

The access to the quiz was:	
really fast	45
fast	77
ok	57
slow	16
way too slow	10

The technical handling of the quiz was:	
really easy	113
easy	63
ok	24
difficult	6
very difficult	0

Did the quiz make the lecture more interesting?	
yes	149
a bit	63
maybe	10
no	3
not at all	0

Did the quiz help you to understand the lecture content?	
yes	75
a bit	98
maybe	36
no	16
not at all	0

Did the quiz increase your motivation?	
yes	49
a bit	74
maybe	42
no	55
not at all	6

Do you want to have such quizzes more often in lecture?	
yes	155
maybe	58
no	14
	227

FIGURE B.3: Students' answers in the survey about the *MobileQuiz* in economics in 2012. (2/2)

Comments
Access only worked by typing in the URL, QR Code Reader did not work
I like
Problems with the display of the Quiz
Great Idea
via link
poor QR code reading ability
often QR code too small to read, Internet connection is bad
internet connection noct working, if worked anything was great
QR too small for backrows, qr scans cannot zoom
good job anyway
prima super
:-)
going via link
thanks!
The Qr don't seem to work out, i assume because of the resolution of the beamer, it works on the sideboard for some reason.
No Problem with the quiz, well done
Have a smarthone but could not connect
Not all quizzes are such a good „complementary“ for the lecture content
Scanning of Barcode did not workl for me, too small, access via url adress, but it was ok
The first lines the QR Code was too small, but xxxx it works perfectly
Very good idea makes stuff interesting
good idea
only front camera
nice idea
Enlarge the QR-Code
I dont like being forced to use this kind of technology because my devices for this are terrible.
used the URL
cool programming
Genereally the wifi on my phone does not work well at the university
Cam did not work, typed URL
Scanning of the QR-Code doesn't always work :-{
The QR Code did not work but typing the URL
QR Code does not work on big white screen, not flat enough,works better on sid board or zoomed in
The Qr-Codes are sometimes hard to read, depending on where you sit
really good
i had to type in the URL
slow loading of the page, equally often is fine
Sometimes it was faster to boot my Pc and go to the URL manually than some people using the QR code
It does not work very well at an angle (QR scanner)
Although i didn't participate myself in the quiz. I followed the solution in the lecture. Good thing those quizzes. I like them. I don't own a smartphone and could not bring my laptop
I used the browser
I did not use the QR-code, I entered the tiny URL manually
Habe versucht die angegebene URL mit meinem Blacberry Bold aufzurufen, was leider nicht funktioniert hat.
it was hard to put in the numbers using the bar or typing bc the app was superslow

FIGURE B.4: Students' comments in the survey about the *MobileQuiz* in economics in 2012. (1/2)

WiFi does not work. Did not work with scanner
I do not have a smartphone and my laptop is very / too big to carry it with me
in the middle of class, we normally participate in quiz might be disturbing
do not own a smartphone. Laptop too complicated to bring with for only this
great idea :)
nice interruptions
used the internet, not the QR-code
1,2 times like now is just enough, no more quizzes are necessary
really cool idea
NO
good thing
good job
questions and answer possibilities should be also presented on the slide and not only in the quiz
I don't own a smartphone, only laptop
sometimes it is difficult to scan if you seat in the back rows
xxx manually
QR code only worked for the seats in the very front
Während der ersten beiden Veranstaltungen kein Zugriff mit Ipad möglich, später keine Probleme. Link eingetippt
nice idea!congrats
good idea!
Mostly, bar codes projected on the wall are too small to be read by a bar code reader on the mobile phone
WiFi does not work on my phone
i do not have a barcode reader
QR-code not readable in many cases
nice job.
more quizzes if it makes sense in the particular context
Solutions should be provided, quiz should be announced to think about it
via smartphone of another person (worked together)
typed in the address
Sometimes it does not work at all!

FIGURE B.5: Students' comments in the survey about the *MobileQuiz* in economics in 2012. (2/2)

Ergebnis Befragung Studierende Seminar "Einführung in die Literaturwissenschaft" (Odermatt)
N=20

Ich habe am Mobile Quiz teilgenommen	
Ja, mit Smartphone	14
Ja, Laptop/Tablet	3
Ja, gewöhnliches Handy	
Nein, funktionierte nicht	2
Nein, zu umständlich	
Nein, Gerät nicht dabei	
Nein, besitze kein Gerät	2

Meist QR-Code oder URL verwendet?	
QR	10
URL	5

Wie schnell konnte der Zugang hergestellt werden?	
sehr schnell	0
schnell	3
ok	8
langsam	5
sehr langsam	1

Technische Handhabung war	
sehr leicht	6
leicht	8
ok	2
schwierig	0
sehr schwierig	1

War die Veranstaltung durch Einsatz des MobileQuiz interessanter?	
stimme voll zu	7
stimme eher zu	4
teils/teils	3
stimme eher nicht zu	4
stimme gar nicht zu	1

Hat das MQ Ihnen geholfen, den Stoff der Veranstaltung besser zu verstehen? (z. B. indem Sie Ihr Verständnis des Stoffs überprüfen konnten?)	
stimme voll zu	2
stimme eher zu	4
teils/teils	6
stimme eher nicht zu	5
stimme gar nicht zu	2

Hat das Mobile Quiz Ihre Motivation erhöht den Stoff der Veranstaltung zu lernen?	
stimme voll zu	1
stimme eher zu	4
teils/teils	6
stimme eher nicht zu	4
stimme gar nicht zu	4

FIGURE B.6: Students' answers in the survey about the *MobileQuiz* in German literature in 2012. (1/2)

Hatten Sie den Eindruck, dass die MobileQuiz-Aufgaben Ihnen geholfen haben, sich im Stoff zu orientieren, obwohl Sie dabei falsche Antworten eingeloggt haben?	
stimme voll zu	1
stimme eher zu	5
teils/teils	10
stimme eher nicht zu	2
stimme gar nicht zu	1
Würden Sie grundsätzlich den Einsatz von MobileQuiz-Aufgaben in Vorlesungen und Seminaren begrüßen?	
Ja	10
weiß nicht	4
nein	3
Anmerkungen/Anregungen:	
zu umständlich für seine Zwecke (P19)	
teilweise technische Schwierigkeiten (P20)	
Anmerkungen zu den Fragebögen:	
Studierende füllen auch aus, wenn sie nicht mitmachen konnten (fanden es interessant und nützlich) - P07 P16	
Bei QR-Code oder URL wurde häufig beides angekreuzt (und dann nicht gewertet)	
Studierende, die kein Smartphone hatten, haben sich mit anderen zusammengetan - P07 P18	

FIGURE B.7: Students' answers in the survey about the *MobileQuiz* in German literature in 2012. (2/2)

Appendix C

Survey for the students about the *HomeQuiz* in 2013

The next eight pages show the results of the two surveys about the *HomeQuiz* in the Spring semester 2013. The first survey was answered by students of the course in computer science (Prof. Effelsberg), the second one by students of German literature (Odermatt).

HDZ

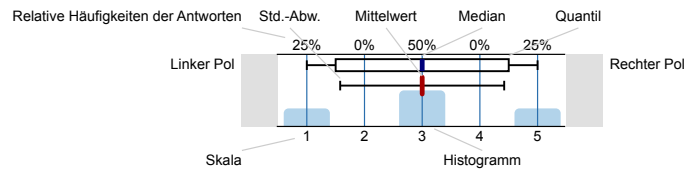
Mobile Quiz mit Feedback - Prof. Effelsberg ()
Erfasste Fragebögen = 46



Auswertungsteil der geschlossenen Fragen

Legende

Fragetext



n=Anzahl
mw=Mittelwert
md=Median
s=Std.-Abw.
E.=Enthaltung

Questions on MobileQuiz

1. Did you use the quiz?



n=46

2. Did you try to answer all questions?



n=44

3. How was the technical handling of the quiz?



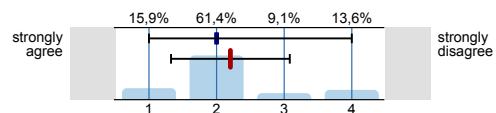
n=43

4. If there were technical problems, which ones occurred?



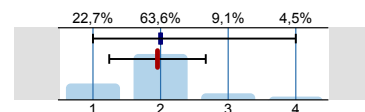
n=46

5. Did the quiz increase your motivation to prepare (or respectively rework) the course?



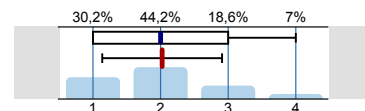
n=44
mw=2,2
md=2
s=0,9

6. Do you think the quiz made the preparation / rework of the course more interesting?



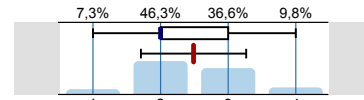
n=44
mw=2
md=2
s=0,7

7. Did the quiz help you to recognize your deficits in understanding the course contents?



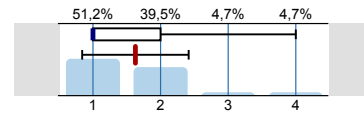
n=43
mw=2
md=2
s=0,9

8. Do understanding deficits, that occurred during the quiz use, motivate you to ask the lecturer in the course?



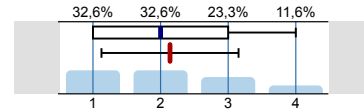
n=41
mw=2,5
md=2
s=0,8

9. In general, do you think the quiz should be used more often for the preparation/rework of courses?



n=43
mw=1,6
md=1
s=0,8

10. Would you prefer more questions in the single quiz rounds?
(e. g 8 questions per quiz instead of 5 questions per quiz)



n=43
mw=2,1
md=2
s=1

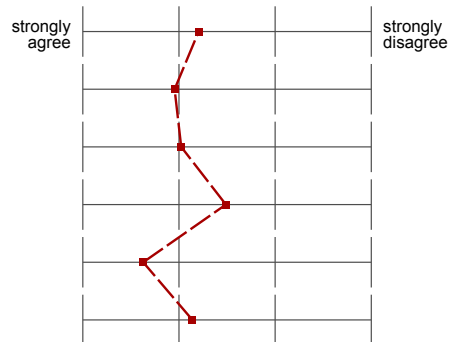
Profillinie

Teilbereich: Referat - Hochschuldidaktikzentrum
 Name der/des Lehrenden: HDZ
 Titel der Lehrveranstaltung: Mobile Quiz mit Feedback - Prof. Effelsberg
 (Name der Umfrage)

Verwendete Werte in der Profillinie: Mittelwert

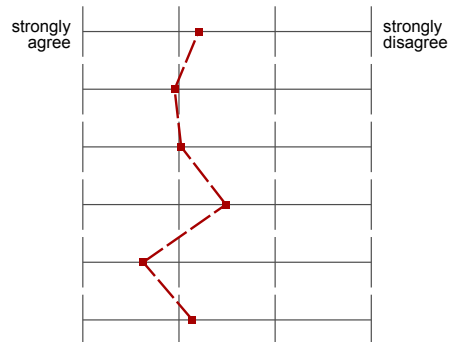
Questions on MobileQuiz

5. Did the quiz increase your motivation to prepare (or respectively rework) the course?



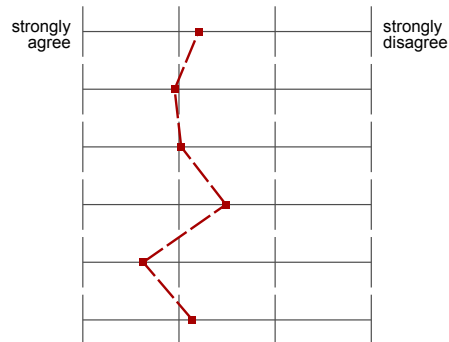
n=44 mw=2,2 md=2,0 s=0,9

6. Do you think the quiz made the preparation / rework of the course more interesting?



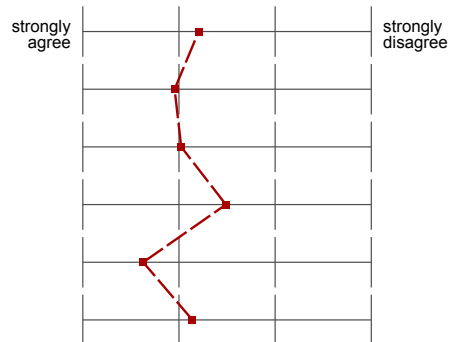
n=44 mw=2,0 md=2,0 s=0,7

7. Did the quiz help you to recognize your deficits in understanding the course contents?



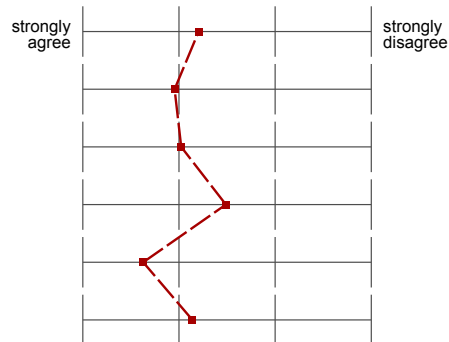
n=43 mw=2,0 md=2,0 s=0,9

8. Do understanding deficits, that occurred during the quiz use, motivate you to ask the lecturer in the course?



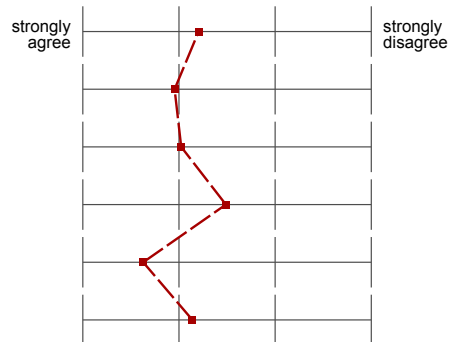
n=41 mw=2,5 md=2,0 s=0,8

9. In general, do you think the quiz should be used more often for the preparation/rework of courses?



n=43 mw=1,6 md=1,0 s=0,8

10. Would you prefer more questions in the single quiz rounds? (e. g 8 questions per quiz instead of 5 questions per quiz)



n=43 mw=2,1 md=2,0 s=1,0

Auswertungsteil der offenen Fragen

Questions on MobileQuiz

11. Do you have further comments, ideas or improvement proposals?

- A quiz should cover a particular topic in one quiz.
- Es wäre schön, wenn man sieht, wie viele wo auch falsch oder richtig lagen/waren! --> Vergleich zwischen Studierenden
- I think in the combination of the normal exercises, the Quiz is good.
- I was unaware of the quiz.
- No idea, thanks.
- Send results by email or publish them at some point.
- When I finished the test, I cannot come back to print the screen later. It's good to lock for new answer, but at least please show the blank questions.

HDZ

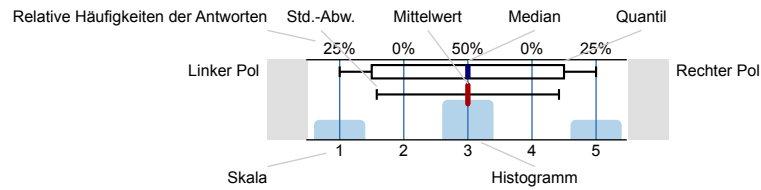
Mobile Quiz mit Feedback - Odermatt ()
Erfasste Fragebögen = 14



Auswertungsteil der geschlossenen Fragen

Legende

Frage-Text



n=Anzahl
mw=Mittelwert
md=Median
s=Std.-Abw.
E.=Enthaltung

Fragen zum MobileQuiz

1. Haben Sie das Quiz bearbeitet?



2. Haben Sie versucht, alle Aufgaben zu lösen?



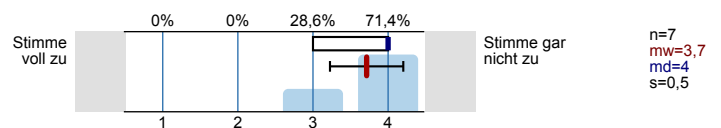
3. Konnten Sie das Quiz technisch durchführen?



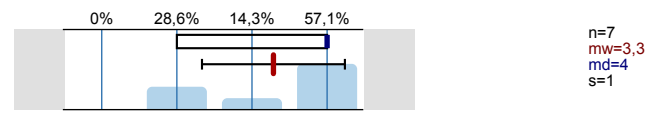
4. Falls Sie technische Probleme hatten, welche waren diese?



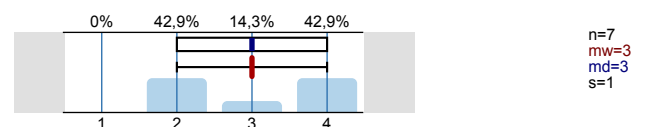
5. Hatten Sie durch das Quiz mehr Motivation, die Lehrveranstaltung vor- bzw. nachzubereiten?



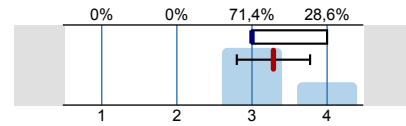
6. Haben Sie das Gefühl, das Quiz hat die Vor- bzw. Nachbereitung interessanter gemacht?



7. Hat Ihnen das Quiz geholfen, zu sehen, wo Sie noch Verständnislücken haben?

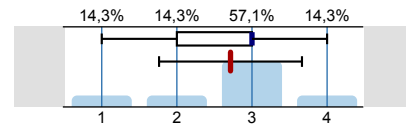


8. Animieren Sie Verständnisprobleme, die Sie im Quiz-Einsatz feststellen, in der Lehrveranstaltung nachzufragen?



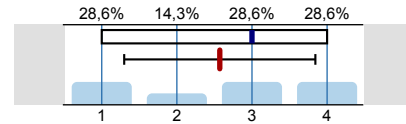
n=7
mw=3,3
md=3
s=0,5

9. Würden Sie sich wünschen, dass das Quiz häufiger in Lehrveranstaltungen für die Vor- und Nachbereitung verwendet wird?



n=7
mw=2,7
md=3
s=1

10. Würden Sie sich für zukünftige Quiz-Runden mehr Fragen zur Vor- und Nachbereitung wünschen? (z. B. statt 5 Fragen pro Quiz lieber 8 Fragen pro Quiz)



n=7
mw=2,6
md=3
s=1,3

Profillinie

Teilbereich: Referat - Hochschuldidaktikzentrum
 Name der/des Lehrenden: HDZ
 Titel der Lehrveranstaltung: Mobile Quiz mit Feedback - Odermatt
 (Name der Umfrage)

Verwendete Werte in der Profillinie: Mittelwert

Fragen zum MobileQuiz

5. Hatten Sie durch das Quiz mehr Motivation, die Lehrveranstaltung vor- bzw. nachzubereiten?

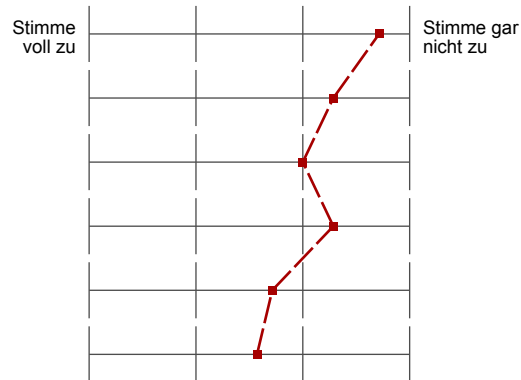
6. Haben Sie das Gefühl, das Quiz hat die Vor- bzw. Nachbereitung interessanter gemacht?

7. Hat Ihnen das Quiz geholfen, zu sehen, wo Sie noch Verständnislücken haben?

8. Animieren Sie Verständnisprobleme, die Sie im Quiz-Einsatz feststellen, in der Lehrveranstaltung nachzufragen?

9. Würden Sie sich wünschen, dass das Quiz häufiger in Lehrveranstaltungen für die Vor- und Nachbereitung verwendet wird?

10. Würden Sie sich für zukünftige Quiz-Runden mehr Fragen zur Vor- und Nachbereitung wünschen?



n=7 mw=3,7 md=4,0 s=0,5

n=7 mw=3,3 md=4,0 s=1,0

n=7 mw=3,0 md=3,0 s=1,0

n=7 mw=3,3 md=3,0 s=0,5

n=7 mw=2,7 md=3,0 s=1,0

n=7 mw=2,6 md=3,0 s=1,3

Appendix D

Survey for the lecturers about the *MobileQuiz* in 2013

Use of MobileQuiz – Questionnaire for Lecturers

1. Preparation of the MobileQuiz

- o What do you think about the **technical** effort during the preparation?
(quiz set up in ILIAS)?
 - too much effort
 - high effort
 - small effort
 - almost no effort
- o What do you think about the **contentual** effort during the preparation?
(preparing the questions)?
 - too much effort
 - high effort
 - small effort
 - almost no effort

Comments:

2. Execution of the MobileQuiz

- o What do you think about the **technical** effort during the execution?
(execution of the quiz within the lecture)?
 - too much effort
 - high effort
 - small effort
 - almost no effort
 -
- o What do you think about the **contentual** effort during the execution?
(explanation of questions & discussion of answers)?
 - too much effort
 - high effort
 - small effort
 - almost no effort
- o How big was the effort to explain the mechanics of the quiz to the students
(e.g. How does the QR-code work? Where do I get a QR-code reader)?
 - too much effort
 - high effort
 - small effort
 - almost no effort
- o In my opinion, regarding the benefit I get from the quiz, the **technical** effort is:
 - ok
 - too high

FIGURE D.1: Survey for the lecturers about the *MobileQuiz*. (1/3)

- o In my opinion, regarding the benefit I get from the quiz, the **contentual** effort is:
- ok
 - too high

Comments:

3. Acceptance of students

- o How do you rate the **acceptance** of the students?

- o Do you think the students' **motivation to participate** in the lecture raises?

- o Do you think the students' **interest in the content** of the lecture raises?

- o Do you think the quiz raises the students' **understanding of the lecture content**?

4. Benefit for you as lecturer

- o Do you believe that you get a better understanding about the knowledge of your students?

- o Which experiences did you make, which you did not expect this way?

- o What would you do differently the next time?

- o Do you want to use the MobileQuiz again? If not, why?

FIGURE D.2: Survey for the lecturers about the *MobileQuiz*. (2/3)

- o What do you wish from MobileQuiz? What do you want to do what isn't possible yet?

5. Conclusion

- o Would you recommend the MobileQuiz to other lecturers?

- o Do you have cardinal interest in exchanging information with other lecturers about the experiences with MobileQuiz?

- o Free Comments:

Thank you for your answers!

Daniel Schön & Melanie Klinger

FIGURE D.3: Survey for the lecturers about the *MobileQuiz*. (3/3)

What do you think about the technical effort during the preparation?			
too much	high	small	no
0	0	4	3

What do you think about the contentual effort during the preparation?			
too much	high	small	no
0	2	4	1

What do you think about the technical effort during the execution?			
too much	high	small	no
2	1	4	2

What do you think about the contentual effort during the execution?			
too much	high	small	no
0	0	5	2

How big was the effort to explain the mechanics of the quiz to the students?			
too much	high	small	no
0	0	2	5

In my opinion, regarding the benefit I get from the quiz, the technical effort is:			
ok	zu hoch		
7	0		

In my opinion, regarding the benefit I get from the quiz, the contentual effort is:			
ok	too high		
6	1		

FIGURE D.4: Results of the survey for the lecturers about the *MobileQuiz*.

Preparation of the MobileQuiz
Es wäre gut wenn der QR tag schon vorher feststeht, so dass man ihn auf die Folien drucken kann
Durch die Fachkultur in der Literaturwissenschaft mussten die Fragen oft sehr komplex und umständlich formuliert werden, so dass ein eindeutiger Wahrheitwert für die Antworten gegeben werden konnte.
Man muss sich gut überlegen, wozu man das Quiz verwenden will und was mit einer Gruppe von Größe X funktioniert. Diese Planung ist nicht ganz trivial.
Execution of the MobileQuiz
Very well accepted.
auf Kürze der URL achten da QR-Code noch nicht zuverlässig
Die Umfrage wurde oft durch zu langsame oder nicht funktionierende Internetverbindungen zu den Telefonen in die Länge gezogen.
Ich empfinde das Quiz als sehr bereichernd, wenn es richtig eingesetzt wird.
How do you rate the acceptance of the students?
99,00%
hoch
Ich glaube, sie benutzen es gerne.
willkommene Abwechslung, wird wohlwollen angenommen
hoch
In den meisten Fällen eine sehr hohe Akzeptanz, in Stunden ohne Quiz wurde sogar danach gefragt.
Hoch/Gut
Do you think the students' motivation to participate in the lecture raises?
80,00%
ja
Ja, die Studenten scheinen sich ernsthaft Gedanken über die gestellten Fragen zu machen.
ja, vor allem für schüchterne Studierende
ja
Nein.
Ja, zu einem gewissen Grad.
Do you think the students' interest in the content of the lecture raises?
80,00%
ja
Eher geringes Erhöhen des Interesses. Das Interesse muss da sein. Das MobileQuiz erhöht lediglich die Motivation mitzumachen.
Die Motivation eine Veranstaltung zu besuchen hat meist andere Gründe (Thema, leicht verdiente Noten, Interesse etc.)
nicht zwangsläufig
Nein.
Ja, kann ich mir vorstellen.
Do you think the quiz raises the students' understanding of the lecture content?
90,00%
ja
Ja, da ich glaube, dass sich die Inhalte besser im Kopf verankern, wenn man mal darüber nachgedacht hat.
nein
Nein.
Nein.
Ja, bei geeigneten Quiz-Inhalten.

FIGURE D.5: Comments of the survey for the lecturers about the *MobileQuiz*. (1/3)

Do you believe that you get a better understanding about the knowledge of your students?
Yes
ja
Ja..... erschreckend / ernüchternd... :)
ja, Wechsel der Fragemodalität erhöht die Aufmerksamkeit für einige Zeit.
ja
Ja, aber eingeschränkt auf die sehr spezifischen Fragen; nicht sehr repräsentativ.
Ein wenig. Das Quiz erlaubt es natürlich, den Wissensstand abzufragen. Allerdings ist es m.E. nicht möglich, tiefergehendes Verständnis oder bestimmte Fehlinterpretationen auf Seiten der Studierenden zuverlässig zu überprüfen. Dafür ist das Quiz einfach zu kurz.

Which experiences did you make, which you did not expect this way?
ich hätte wirklich nicht gedacht, daß es so einfach ist!
Dass doch nur die Hälfte zuhört und den Inhalt versteht.
Dass das Einlesen des QR-Codes noch nicht so zuverlässig funktioniert wie erwartet. Das könnte sich mittlerweile schon verbessert haben.
keine
Es ist sehr leicht und schnell eingerichtet und sorgt für große Unterhaltung und eine absolut freudige Resonanz bei den Studierenden.
Eigentlich gab es nichts gänzlich Unerwartetes. Ich hätte fast noch mehr Beteiligung erwartet (mehr Smart-Phones o.ä.).

What would you do differently the next time?
I would not mix different topics.
es u. U. Häufiger nutzen
Nichts
Evtl. auf mehrere Veranstaltungen ausdehnen und enger mit dem Aufbau des Kurses verknüpfen. Am Besten schon bei der Planung einer neuen Vorlesung das Tool mit einbeziehen.
vorher schon das tag auf die Folien drucken, da es aus den letzten Reihen nicht gelesen werden konnte
Ich müsste die Fragen sehr viel intensiver vorbereiten und einen Weg finden für die Lernziele des Fachs angemessen zu formulieren.
Noch etwas mehr Wissensabfragen (zusätzlich zu Entscheidungsexperimenten).

Do you want to use the MobileQuiz again? If not, why?
Yes
ja
Jep
auf jeden Fall – if applicable.
ja
Ich werde ihm sicher noch eine Chance geben, habe aber die Befürchtung, dass es sich für meine Lehrform (geisteswissenschaftliches PS mit ca. 20 TN) letztlich nicht so gut eignet.
Ja.

What do you wish from MobileQuiz? What do you want to do what isn't possible yet?
I would like to use tex to make formulas.
Das Tool genügt meinen Ansprüchen.
Rückmeldung über eingegangene Ergebnisse (zumindest die Anzahl) in Echtzeit und fortlaufend.
s.o
Sehr hilfreich wäre eine leistungsstarke Einbindungsmöglichkeit für Grafiken.
Anonymisiertes Tracking von Individuen über verschiedene Quiz-Fragen oder Quizze hinweg.

FIGURE D.6: Comments of the survey for the lecturers about the *MobileQuiz*. (2/3)

Would you recommend the MobileQuiz to other lecturers?
Yes, I will have a workshop to show it.
ja
Ja, auf jeden Fall.
kann jeder benutzen, auch mit wenig technischem Vorwissen.
ja
Auf jeden Fall, vor allem, wenn es sich um Massenveranstaltungen mit "positivem Wissen" (hard sciences) handelt.
Ja, auf jeden Fall. Habe ich auch schon getan.

Do you have cardinal interest in exchanging information with other lecturers about the experiences with MobileQuiz?
How to make it more useful.
nein
Geht so... =P
kein Interesse einen aktiven Austausch anzustossen aber ich würde mich einer Diskussion nicht verschließen.
nein
Gerne.
Ja.

Free Comments:
??
Vielleicht wären Kurse zu den neuen Möglichkeiten die Lehre technisch zu unterstützen auch ein Thema für das HDZ.
Das Quiz ist imGrunde ein tolles Tool, das ich noch nicht ganz auf meine Bedürfnisse habe abstimmen können!

FIGURE D.7: Comments of the survey for the lecturers about the *MobileQuiz*. (3/3)

Appendix E

Survey about the *MobileQuiz2* in the Fall semester 2015

For which purposes did you use the MobileQuiz2 in your teaching context?	
short break for students in a ninety minute lecture.	
Knowledge about learning success of students.	
To give the students an opportunity to work on exercises and to reduce boredom in the lecture.	
In your opinion, how can student learning benefit from using the MobileQuiz2?	
They recognize if they really understood the topic.	
They have to think about a topic and have to apply their knowledge to the questions.	
Topics not yet discussed within the lecture may be discussed after the quiz.	
By reflecting on the presented knowledge, they are able to memorize it better. The increased interactivity makes the lecture more entertaining.	
What do you think is the greatest challenge when using MQ2?	
Finding suitable questions.	
It's time consuming to prepare the questions for the first time.	
It's difficult to come up with a new and good teaching scenario without any skeleton.	
Thinking of the right scenario and questions. The quiz failing is a challenge if the structure of the lecture depends on it.	
Do you think that MQ2 can – after an introduction – be used by lecturers without technical or didactical support? If not, what kind of support is necessary?	
It is usable after an introduction.	
Due to the flexibility of the quiz a contact person for upcoming questions is necessary.	
For some of the scenarios (e.g. twitter wall) some guidelines with "best practices" would be helpful. The plain quiz is easy enough to integrate into the lecture.	
What would you say about students' opinions on the MQ2? Is there a difference in acceptance/opinion in contrast to classical automatic response systems (like the MobileQuiz1 ILIAS-plugin)?	
Students like the quiz.	
The quizzes are strongly recommended by the students.	
I think the students enjoy using their own smartphones to take part in the quiz. That the quiz is working properly is also important for the acceptance. Additional scenarios bring more variety and interaction into the lecture.	
What should be considered when MQ2 will be enhanced and improved?	
More predefined blueprints of scenarios.	
Technical reliability should be improved.	
permanent contact person should be available.	
More scenarios with best practices.	
What kind of teaching/learning scenario would you like to create and use?	
Some interactive scenarios with sorting of results.	
-	
Is there any additional kind of analysis you would like to conduct with students' responses?	
-	
I'm satisfied with the information that is given by the mobilequiz2. it is important for me to know when the students are done with the quiz.	
How important is it to you that the system is adaptable to your individual needs?	
not important	0
...	1
...	0
very important	1
How important is it for you to have an overview of students' activities in the system? (The GameMaster Page) (1 = not important 4 = very Important)	
not important	0
...	0
...	2
very important	0
Could you also think of using MQ2 for research purposes?	
no.	
no	

FIGURE E.1: Answers of the evaluation of the customization of the *MobileQuiz2* in the Fall semester 2015 given by lecturers who did use the tool regularly. (1/2)

In your opinion, how technically oriented is your current teaching and research discipline?	
very technically oriented	2
...	0
...	0
absolutely not technically oriented	0
How would you describe yourself personally?	
very technically oriented	2
...	0
...	0
absolutely not technically oriented	0
How would you describe yourself concerning your teaching?	
very committed	1
...	1
...	0
absolutely not committed	0
Free Comments	
.	

FIGURE E.2: Answers of the evaluation of the customization of the *MobileQuiz2* in the Fall semester 2015 given by lecturers who did use the tool regularly. (2/2)

Why haven't you used MobileQuiz2 in a lecture yet?	
Well, I've used MobileQuiz2.	
End of semester time constraints. More importantly, the design of the course did not include the tool as it was introduced to me during the semester.	
I have already used MobileQuiz2. It should be expected. Students must be technically ready for that :-)	
Do you intend to use MQ2 in your teaching context?	
If yes, for which purposes? If not, why not?	
Yes, for giving possibility to learn while testing students and for giving me feedback from them.	
Yes. Include interactive elements as advanced organizer at the beginning of a lecture, as response to tasks during the lecture or as summary at the end of the lecture.	
Yes. To check student's understanding during lectures, to help them revision. Lecturer knows which part of knowledge is difficult.	
In your opinion, how can student learning benefit from using the MobileQuiz2?	
They are more aware of what they know and what not, since everyone answers questions by himself.	
As I said before, they can even learn during the quiz...	
See my comments how I would use the tool: Activating prior knowledge, facilitate learning processes, reflect on learning processes.	
MobileQuiz2 enables quick revision. With applets - students can test out new knowledge, substantially.	
What do you think is the greatest challenge for the lecturer when using MQ2?	
Getting everyone participating	
The possibility of using math-expressions and GG applets.	
Introducing the tool for seamless integration in the classroom.	
Create wise questions....	
After an introduction to MQ2, do you think that it can be used by lecturers without further technical or didactical support?	
If not, what kind of support is necessary?	
Yes	
Yes	
As with all learning material, information is required for successful use.	
The lecturer needs easy and as quick as possible tool to build a test.	
What would you say about students' opinions on the MQ2? Is there a difference in acceptance/opinion in contrast to classical automatic response systems (like the MobileQuiz1 ILIAS-plugin)?	
I don't know	
--	
N/A	
Yes, there is a difference. The math text looks professional and possibility of adding applets is priceless.	
What should be considered when MQ2 will be enhanced and improved?	
Make the GUI more userfriendly for teachers	
N/A	
speed of working application (loading objects)	
What kind of teaching/learning scenario would you like to create and use?	
N/A	
with html, question_single, question_multi, OpenQuestion, image, comment, LikertQuestion	
Is there any additional kind of analysis you would like to conduct with students' responses?	
N/A	
I know, that's not the idea of the MQ2. Anyway, could it be possible to see answers of a certain person? For example, in a kind of OpenQuestion there can be given a students name... Is it possible to see this record of answers? I realize it depends on the way of storage the data and probalby it's not worth/sense of changing anything. Just asking :-)	
It could be good to give an award to the student, who was the best.	
How important is it to you that the system is adaptable to your individual needs?	
not important	0
...	0
...	2
very important	2

FIGURE E.3: Answers regarding the customization of the *MobileQuiz2* in the Fall semester 2015 given by lecturers who did not use the tool regularly. (1/2)

How important is it for you to have an overview of students' activities in the system? (The GameMaster Page) (1 = not important 4 = very Important)	
not important	1
...	1
...	1
very important	1
Could you also think of using MQ2 for research purposes?	
Yes	
Yes.	
Yes.	
In your opinion, how technically oriented is your current teaching and research discipline?	
very technically oriented	0
...	1
...	2
absolutely not technically oriented	1
How would you describe yourself personally?	
very technically oriented	0
...	0
...	2
absolutely not technically oriented	2
How would you describe yourself concerning your teaching?	
absolutely not committed	0
...	0
...	1
very committed	3
Free Comments	
Thank you :-)	

FIGURE E.4: Answers regarding the customization of the *MobileQuiz2* in the Fall semester 2015 given by lecturers who did not use the tool regularly. (2/2)

The system creates variation that makes the course more interesting.	
absolutely disagree	2
...	2
...	8
absolutely agree	15
Using the system is fun.	
absolutely disagree	2
...	6
...	9
absolutely agree	10
Using the system helps me to understand the course contents.	
absolutely disagree	0
...	9
...	13
absolutely agree	5
Using the system distracts me more than it helps me to understand the course contents.	
absolutely disagree	15
...	7
...	4
absolutely agree	1
It is important that the system can be used anonymously.	
absolutely disagree	0
...	5
...	5
absolutely agree	17
I would like lecturers to use the system regularly.	
absolutely disagree	1
...	6
...	14
absolutely agree	6
I would also accept to install a specific program to use the system.	
absolutely disagree	8
...	10
...	6
absolutely agree	3
I would like to get grades on the answers I give via the system.	
absolutely disagree	13
...	8
...	5
absolutely agree	1
Here you can leave a free comment:	
There are to many bugs in it.	
Please try to make the Quiz running stable. :)	

FIGURE E.5: Students' answers regarding the customization of the *MobileQuiz2* in the Fall semester 2015.